1990年5月1日発行（毎月1回1日発行）第9巻5号通巻97号　昭和58年11月2日第三種郵便物認可

特集 BASICプログラミング入門

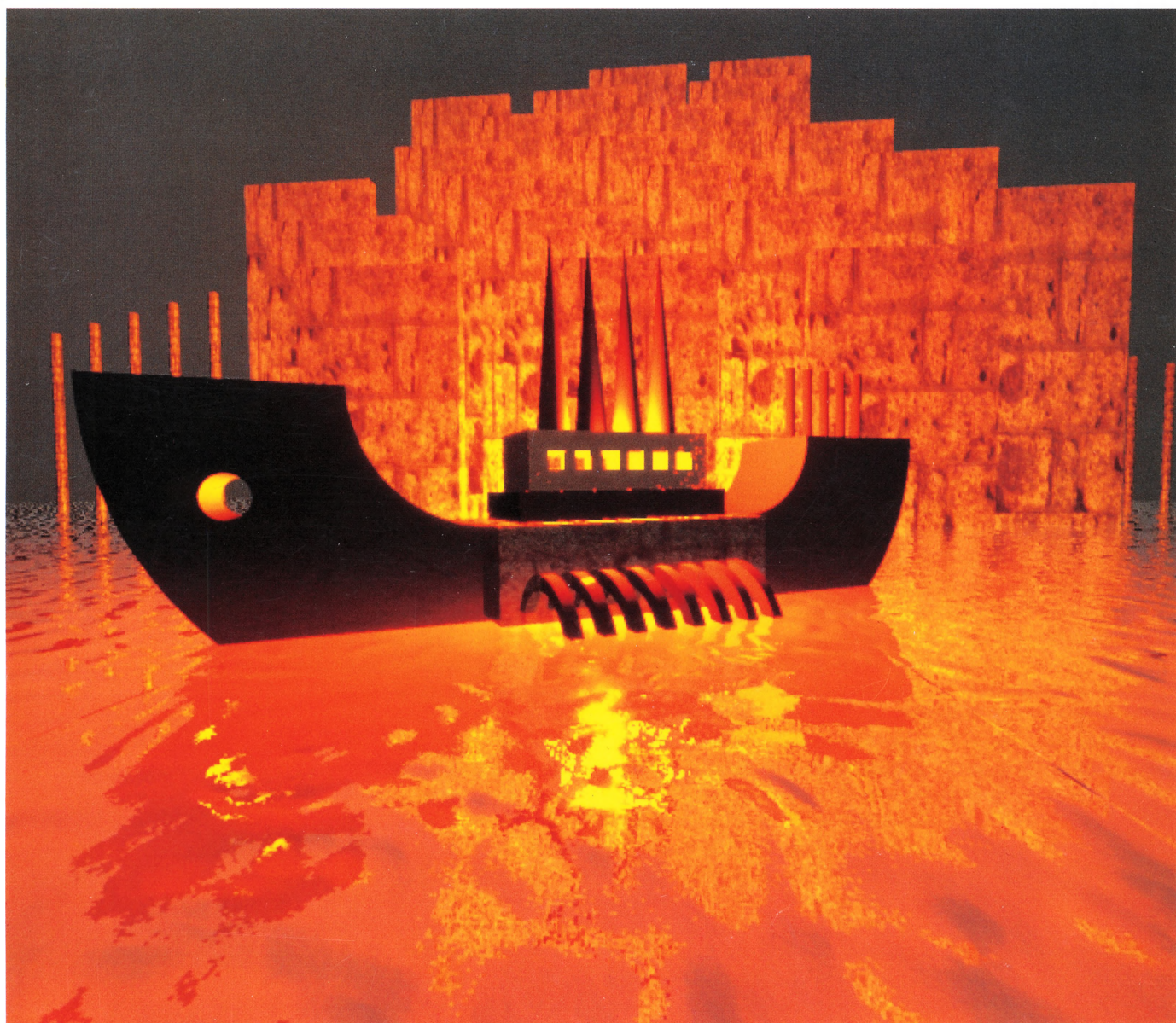
第5回 言わせてくれなくちゃだワ

新製品 X68000 PROⅡ/EXPERTⅡ/SUPER

これがSX-WINDOWだ!

5
1990

オー！エックス
定価560円

SHARP New-Life People

X68000
SX-WINDOW

ひらかれた知性。

見はてぬ夢の象徴。

次代のインテリジェンス、"SX-WINDOW"搭載。

●いま、こだわり続けてきたある執着がまさに帰結しようとしています。グラフィカルユーザーインターフェイス"SX-WINDOW ver.1.0"。もちろん、X68000には発売当初よりビジュアルシェルが搭載されていたことはご存じのとおりですが、クオリティグラフィックやマルチメディア、マルチタスク対応など真の意味での汎用性を志向した開発コンセプトからは、私たち自身ものたりなさを禁じ得なかったことも事実です。しかし、キャラクタユーザーインターフェイス全盛のその時代に、デスクトップの概念をいち早く採り入れた先見性は、現在のインターフェイスの在り方に対する的確な予測に基づくもので、何よりも、トレンディなユーザーの圧倒的な支持によって証明されています。パーソナルコンピュータがその意味どおり、個人のためのツールなら、インターフェイスの発展は必然です。このウインドウシステムは、私たちX68000開発プロジェクトに携わったすべてのスタッフの指標であり、義務でもあったのです——。

●ユーザー本位の操作環境を提供するフル画面マルチウインドウタイプのデスクトップ(テキスト面/単色4階調＋単色カラー4色。グラフィック面/カラー65,536色中16色)、新感覚スクロールバー…こだわりの美学で高められたユーザーインターフェイス。イベント・ドリブン型マルチタスク処理により複数の作業を同時に処理できる疑似マルチタスクや入出力装置の設定が簡単におこなえる多機能コントロールパネルを搭載した本格ウインドウシステムです。

●"SX-WINDOW"、このひらかれた知性は、今もそしてすぐ後に続く時代をも包含した質の高い「愉しみ」を提供するインターフェイスです。フレンドリーOS Human68kはここに、当初の目的の成就と共に、将来へ確かな展望を明示したといえるでしょう。さまざまなジャンルへ、拡がりと密度を高めるアプリケーション環境、インテリジェントなペリフェラル環境。こうしたトレンドを背景に、いま第4世代のX68000がデビューします。

NEW X68000
PERSONAL WORKSTATION

SUPER・EXPERT・PRO

ザ・ワークステーション。80Mバイトハードディスク、SCSIインターフェイスを標準装備。
SUPER HD 本体＋キーボード＋マウス・トラックボール
HDタイプ CZ-623C-TN(チタン) 標準価格498,000円(税別)〈6月発売予定〉

アートの系譜。**EXPERT II** 本体＋キーボード＋マウス・トラックボール
CZ-603C-BK(ブラック)・-GY(グレー) 標準価格338,000円(税別)/HDタイプ CZ-613C-BK(ブラック) 標準価格448,000円(税別)

ニュースタンダード。**PRO II** 本体＋キーボード＋マウス
CZ-653C-GY(グレー)・-BK(ブラック) 標準価格285,000円(税別)
HDタイプ CZ-663C-GY(グレー)・-BK(ブラック) 標準価格395,000円(税別)

15型カラーディスプレイテレビ(ドットピッチ0.39m) CZ-602D-BK(ブラック)・-GY(グレー)……………… 標準価格 99,800円(チルトスタンド同梱・税別)
15型カラーディスプレイテレビ(ドットピッチ0.39m) CZ-605D-BK(ブラック)・-GY(グレー)……………… 標準価格115,000円(スピーカー2個/チルトスタンド同梱・税別)
15型カラーディスプレイテレビ(ドットピッチ0.31m) CZ-613D-TN(チタン)・-BK(ブラック)・-GY(グレー) 標準価格135,000円(スピーカー2個/チルトスタンド同梱・税別)
14型カラーディスプレイ(ドットピッチ0.31m) CZ-603D-BK(ブラック)・-GY(グレー)……………… 標準価格 84,800円(チルトスタンド同梱・税別)
14型カラーディスプレイ(ドットピッチ0.31m) CZ-604D-BK(ブラック)・-GY(グレー)……………… 標準価格 94,800円(スピーカー2個/チルトスタンド同梱・税別)

EXEリーダーズグッズ プレゼント実施中
●いま、EXE会員よりご紹介のお客様がEXEショップでX68000シリーズを購入されますと、EXE会員にEXEリーダーズグッズをプレゼントします。詳しくはEXEショップにお問い合わせください。
●また、X68000シリーズをご購入のお客様は、ぜひEXEクラブにご入会ください。

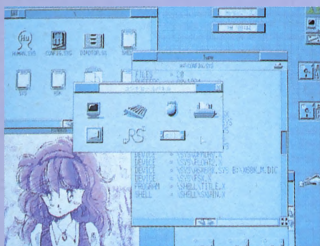
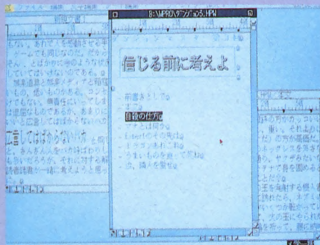
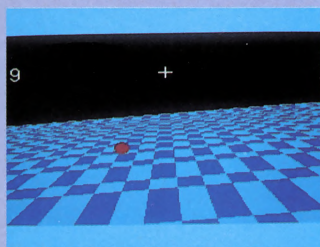
本広告に掲載しております商品および役務の価格には消費税は含まれておりませんので、ご購入の際、消費税額をお支払い下さい。

●お問い合わせは…シャープ㈱電子機器事業本部システム機器営業部〒545大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
電子機器事業本部テレビ事業部第4商品企画部 〒162東京都新宿区市谷八幡町８番地☎(03)260-1161(大代表)
シャープ株式会社

Oh!X

CONT

〈スタッフ〉
●編集長／前田　徹　●編集／植木章夫　太田慎一　岡崎栄子　●協力／有田隆也　中森　章　後藤貴行　林　一樹　荻窪　圭　岡本浩一郎　毛内俊行　吉田賢司　影山裕昭　相馬英智　古村　聡　村田敏幸　丹　明彦　三沢和彦　長沢淳博　宮島　靖　金子俊一　浦川博之　●カメラ／杉山和美　●イラスト／永沢しげる　山田晴久　小栗由香　●アートディレクター／島村勝頼　●レイアウト／元木昌子　AD GREEN　●校正／千野延明　織田洋子

新製品 X 68000SUPER-HD

SX-WINDOW

Hyperword PRO-68K

特集 BASICプログラミング

X68000用CARD.FNC

ラジコンスティック

表紙絵：塚田哲也

1990 MAY 5

E N T S

■広告目次

SHARP

クリエイティブマインドあふれる周辺機器が

ディスプレイ関連

カラーディスプレイテレビ

15型カラーディスプレイテレビ
CZ-602D-GY･-BK
標準価格 99,800円(税別)
(チルトスタンド同梱)

カラーディスプレイ

14型カラーディスプレイ
CZ-603D-GY･-BK
標準価格 84,800円(税別)
(チルトスタンド同梱)

15型カラーディスプレイテレビ
CZ-605D-GY･-BK
標準価格 115,000円(税別)
(スピーカー2個・チルトスタンド同梱)

21型カラーディスプレイ
CU-21HD
標準価格 148,000円(税別)
(スピーカー2個同梱)

15型カラーディスプレイテレビ
CZ-613D-GY･-BK･-TN
標準価格 135,000円(税別)
(スピーカー2個・チルトスタンド同梱)

チューナー

RGBシステムチューナー
CZ-6TU-GY･-BK
標準価格 33,100円(税別)
(リモコン付)

カラーディスプレイ

14型カラーディスプレイ
CZ-604D-GY･-BK
標準価格 94,800円(税別)
(スピーカー2個・チルトスタンド同梱)

CRTフィルター

高性能CRTフィルター
BF-68PRO
標準価格 19,800円(税別)
(14/15型用)

アートツール

画像入力

カラーイメージスキャナ※1
CZ-8NS1
標準価格 188,000円(税別)

スキャナ用パラレルボード
CZ-6BN1
標準価格 29,800円(税別)

映像入力

カラーイメージユニット※2
CZ-6VT1
CZ-6VT1-BK
標準価格 69,800円(税別)

プリンタ

カラープリンタ

24ドット
熱転写カラー漢字プリンタ
CZ-8PC3
標準価格 65,800円(税別)
(信号ケーブル同梱)

48ドット
熱転写カラー漢字プリンタ
CZ-8PC4
CZ-8PC4-GY
標準価格 99,800円(税別)
(信号ケーブル同梱)

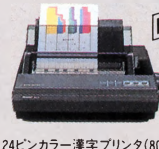
カラービデオプリンタ

カラービデオプリンタ
★CZ-6PV1
標準価格 198,000円(税別)
(信号ケーブル同梱)

カラーイメージジェット

カラー イメージジェット※3
IO-735X
標準価格 248,000円(税別)
(信号ケーブル別売)

ドットプリンタ

NEW

24ピンカラー漢字プリンタ(80桁)
CZ-8PG1
標準価格 130,000円(税別)
(信号ケーブル同梱)

NEW

24ピンカラー漢字プリンタ(136桁)
CZ-8PG2
標準価格 160,000円(税別)
(信号ケーブル同梱)

NEW

24ピン漢字プリンタ(136桁)
CZ-8PK10
標準価格 97,800円(税別)
(信号ケーブル同梱)

ファイル

ハードディスク

ハードディスクユニット(20MB)
CZ-620H
標準価格 178,000円(税別)

増設用ハードディスクドライブ
(40MB)
CZ-64H
標準価格 120,000円(税別)
(取付費別)
※取付に関してはシャープお客様ご相談窓口にてご相談ください。

※1 ご使用に際しては、カラーイメージスキャナCZ-8NS1に同梱のRS-232Cケーブルで接続するか、より高速のパラレルデータ伝送を行う場合、別売のスキャナ用パラレルボードCZ-6BN1標準価格29,800円(税別)で接続してください。
※2 CZ-603D/604D、CU-21HDをご使用の場合は、RGBシステムチューナーCZ-6TU(別売)が必要です。
※3 別売の信号ケーブルIO-73CX標準価格5,500円(税別)で接続して下さい。

X1・X1turbo シリーズ用 周辺機器

標準価格は税別です。

カラーディスプレイ		
●21型カラーディスプレイ※1	CU-21HD	148,000円

映像・画像入力編集装置		
●カラーイメージスキャナ	CZ-8NS1	188,000円
●カラーイメージボードII	CZ-8BV2	39,800円
●立体映像セット	★CZ-8BR1	29,800円
●パーソナルテロッパ※2	CZ-8DT2	44,800円

FM音源		
●ステレオタイプFM音源ボード	CZ-8BS1	23,800円

スピーカー(2本1組)標準装備、ミュージックツール同梱

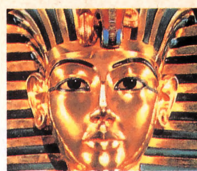
プリンタ		
●24ピンカラー漢字プリンタ(80桁)	CZ-8PG1	130,000円
●24ピンカラー漢字プリンタ(136桁)	CZ-8PG2	160,000円
●24ピン漢字プリンタ(136桁)	CZ-8PK10	97,800円
●24ドット熱転写カラー漢字プリンタ	CZ-8PC3	65,800円
●48ドット熱転写カラー漢字プリンタ	CZ-8PC4	99,800円
●48ドット熱転写カラー漢字プリンタ	CZ-8PC4-GY	99,800円
●カラービデオプリンタ	★CZ-6PV1	198,000円
●カラーイメージジェット	IO-735X	248,000円

ファイル		
●ミニフロッピーディスクユニット(2HD・2D)※3	★CZ-520F	118,000円

資料請求券 X68000 ペリフェラル oh!X 5横

シャープ株式会社

●お問い合わせは…シャープ(株)電子機器事業本部システム機器営業部 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)

X68000をサポート。

シャープペリフェラルファミリー X68000

ボード

拡張メモリ

1MB増設RAMボード
(CZ-600C用)
CZ-6BE1
標準価格 35,000円(税別)

1MB増設RAMボード
(CZ-601C/611C/652C/653C/662C/663C用)
CZ-6BE1B
標準価格 28,000円(税別)

2MB増設RAMボード※4
CZ-6BE2
標準価格 79,800円(税別)

4MB増設RAMボード※4
CZ-6BE4
標準価格 138,000円(税別)

インターフェイス

ユニバーサルI/Oボード
CZ-6BU1
標準価格 39,800円(税別)

GP-IBボード
CZ-6BG1
標準価格 59,800円(税別)

増設用RS-232Cボード
(2チャンネル)
CZ-6BF1
標準価格 49,800円(税別)

数値演算プロセッサ

数値演算プロセッサボード
CZ-6BP1
標準価格 79,800円(税別)

FAX

FAXボード
CZ-6BC1
標準価格 79,800円(税別)

MIDI

MIDIボード
CZ-6BM1
標準価格 26,800円(税別)

ネットワーク

モデム

モデムユニット※5
CZ-8TM2
標準価格 49,800円(税別)
(RS-232Cケーブル同梱)

RS-232Cケーブル

RS-232Cケーブル
(平行接続型)
CZ-8LM1
標準価格 7,200円(税別)

RS-232Cケーブル
(クロス接続型)
CZ-8LM2
標準価格 7,200円(税別)

LANボード

NEW

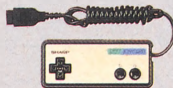
LANボード
CZ-6BL1
標準価格 268,000円(税別)
※電源ユニット・ソフトウェア(ネットワークドライバVer1.0)同梱

入力

インテリジェントコントローラ
CZ-8NJ2
標準価格 23,800円(税別)

マウス・トラックボール
CZ-8NM3
標準価格9,800円(税別)

トラックボール
CZ-8NT1
標準価格 13,800円(税別)

マウス
CZ-8NM2A
標準価格 6,800円(税別)

ジョイカード
CZ-8NJ1
標準価格 1,700円(税別)

その他

拡張スロット

拡張 I/Oボックス(4スロット)
(CZ-600C/601C/611C/602C/612C/603C/613C/623C用)
CZ-6EB1
CZ-6EB1-BK
標準価格 88,000円(税別)

スピーカー

アンプ内蔵
スピーカーシステム(2本1組)
AN-S100
標準価格 36,600円(税別)

システムラック

システムラック
(CZ-600C/601C/611C/602C/612C/603C/613C/623C用)
CZ-6SD1
標準価格 44,800円(税別)

※4 ご使用に際しては、あらかじめ別売の1MB増設RAMボードCZ-6BE1 標準価格35,000円(税別・CZ-600C用)、CZ-6BE1B 標準価格28,000円(税別・CZ-601C、CZ-611C、652C、653C、662C、663C用)を増設してください。
※5 モデムユニットCZ-8TM2に同梱のソフトはX1/X1ターボシリーズ用です。

品名	型名	価格
●ミニフロッピーディスクユニット(2D)	★ CZ-502F	99,800円
●ミニフロッピーディスクユニット(2D・1ドライブ)	CZ-503F	49,800円
●増設用ミニフロッピーディスクドライブ(2D)※4	CZ-53F-BK	19,800円

拡張ボード・その他		
●モデムユニット(300/1200ボー)	CZ-8TM2	49,800円
●320KB外部メモリ	CZ-8BE2	29,800円
●RS-232C・マウスボード※5	CZ-8BM2	19,800円
●フロッピーディスクインターフェイス※6	CZ-8BF1	14,800円
●JIS第1水準漢字ROM※7	CZ-8BK2	19,800円
●RS-232C用ケーブル(平行接続型)	CZ-8LM1	7,200円
●RS-232C用ケーブル(クロス接続型)	CZ-8LM2	7,200円
●拡張 I/Oボックス	CZ-8EB3	33,800円
●RFコンバータ※8	AN-58C	2,980円
●インテリジェントコントローラ	CZ-8NJ2	23,800円
●マウス・トラックボール	CZ-8NM3	9,800円
●マウス	CZ-8NM2A	6,800円
●トラックボール	CZ-8NT1	13,800円
●ジョイカード	CZ-8NJ1	1,700円
●チルトスタンド	CZ-6ST1-E・-B	5,800円
●高性能CRTフィルター ※9	BF-68PRO	19,800円
●スキャナ用パラレルボード ※10	CZ-8BN1	27,800円

●品番中の-表示は、B〈ブラック〉・E〈オフィスグレー〉を示します。※1 X1ターボZシリーズ用 ※2 CZ-862Cには接続できません ※3 X1ターボシリーズ用 ※4 CZ-830C用 ※5 X1シリーズ用 ※6 CZ-850CでCZ-520Fを使用する場合に必要 ※7 CZ-800C、801C、802C、803C、811C、820C用 ※8 CZ-820C、822C、830C用 ※9 14/15型用 ※10 CZ-8NS1用 ●接続等の説明につきましては、周辺機器総合カタログをご参照ください。

★印の商品は在庫僅少です。

本広告に掲載しております商品および役務の価格には消費税は含まれておりませんので、ご購入の際、消費税額をお支払い下さい。

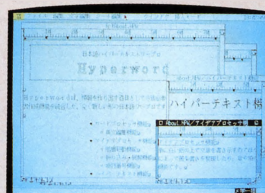
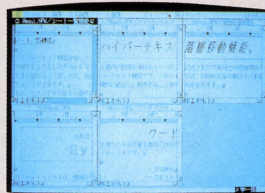
電子機器事業本部テレビ事業部第4商品企画部 〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

SHARP

"アート"と呼べる高水準のソフトウェアが

並のワープロじゃものたりない。アイデアあふれる人の知的ツール、「ハイパーワード」新登場。

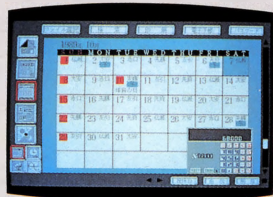
X68000の優れたグラフィック環境をユーザーインターフェースに活用して、効率的に文書を作成するためのインテリジェントツール、それがハイパーワードです。"WYSIWYG"な画面で表現力あふれる文書を作成、印刷できます。アイデアをうまく活用できるアイデアプロセッサ機能、論文やメモ、個人ノートなどを有機的に結合、検索できるハイパーテキスト機能をサポート。データの整理に、プレゼンテーションツールに、単なるワープロを超えた幅広い用途に利用できます。

《ワードプロセッサ機能》●4種類の文字サイズ●9種類の文字修飾、4種類の回転、8種類の下線、8種類の罫線●4種類の割り付け●英文編集機能装備

ハイパーテキストワープロ PRO-68K

Hyperword

CZ-251BS 標準価格39,800円(税別)

情報人の24時間をマネージメント、データと上手につきあえるサイバーノート。

プライベートなデータやビジネスデータを簡単な操作で管理・運営できるパーソナルデータベースです。リフィル、タックシール、ハガキなどへの印字もOK。シャープ電子手帳とのデータ交換(別売の通信ケーブル CE-200Lが必要)も実現。電子手帳をX68000の情報端末として利用できます。

CYBERNOTE PRO-68K

CZ-243BS 標準価格19,800円(税別)

必要なときいつでも使える、メモリ常駐型のステーショナリーツール。

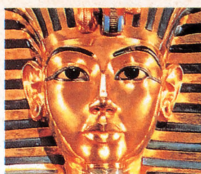
他のソフトを起動する前に、このStationeryPRO-68Kを一度起動するだけ。これで、他のソフトを実行中にも、「メモ」や「スケジュール」、「住所録」など、このソフトが持つ多彩な機能がワンタッチで使えます。またシャープ電子手帳とのデータの送受信も実現(別売の通信ケーブル CE-200Lが必要)。

Stationery PRO-68K

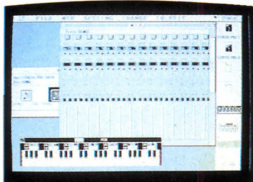
CZ-240BS 標準価格14,800円(税別)

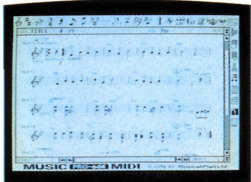
シャープ株式会社 ●お問い合わせは…シャープ(株)電子機器事業本部テレビ事業部 第4商品企画部 〒162 東京都新宿区市谷八幡町8番地 ☎(03)260-1161(大代表)へ。

資料請求券 CZソフト

X68000をサポート。

シャープオリジナルソフトウェア X68000

サウンドツール

Musicstudio PRO-68K ver.1.1
■CZ-252MS 標準価格28,800円(税別)
24トラック対応MIDIマルチレコーディングソフトMusicstudio PRO-68Kがバージョンアップしました。従来の機能に加え、小節間のコピー及びデリートや、MIDIインプットモニターなど、数々の機能を追加・改良。さらに使いやすくなりました。
※MIDIボード(CZ-6BM1)が必要です。

MUSIC PRO-68K〔MIDI〕
■CZ-247MS 標準価格28,800円(税別)
MIDI対応自動伴奏機能をサポート、簡単な楽譜入力で演奏が楽しめます。
※MIDIボード(CZ-6BM1)が必要です。

ソングライブラリ〈101曲集〉
■CZ-248MS 標準価格8,800円(税別)
鑑賞用と音楽データ加工作成用からなるライブラリです。

Sampling PRO-68K
■CZ-215MS 標準価格17,800円(税別)
AD PCM機能を活かす高機能サンプリングエディタ。多彩なEDITORを装備、サンプリング音のデータはBASICでも活用できます。

SOUND PRO-68K
■CZ-214MS 標準価格15,800円(税別)
スタジオのコンソールパネルを操作する感覚でFM音源による音創りが楽しめるサウンドエディタ。

MUSIC PRO-68K
■CZ-213MS 標準価格18,800円(税別)
最大8パートのスコア(総譜)が書け、内蔵のFM音源で演奏できる楽譜ワープロ＆演奏用ツール。

アートツール

NEW PrintShop PRO-68K
■CZ-221HS 標準価格19,800円(税別)
オリジナリティあふれるはがき等、簡単に作成、印刷できるホームプロダクティビリティツール。ほとんどの処理をアイコンで表示しマウスで選ぶフレンドリーオペレーション。

グラフィックライブラリ VOL.1
■CZ-235GS 標準価格8,800円(税別)
暑中見舞用を中心としたNEW Print Shop PRO-68K用グラフィックデータ集。

グラフィックライブラリ VOL.2
■CZ-236GS 標準価格8,800円(税別)
年賀状を中心としたNEW Print Shop PRO-68K用グラフィックデータ集。

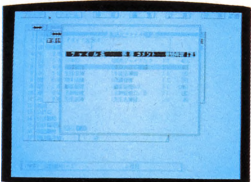
ビジネスツール

TOP給与計算エキスパート
■CZ-228BS 標準価格200,000円(税別)
給与計算から明細発行までを、リアルイメージ入力により自動的に、素早く処理することができます。

TOP財務会計
■CZ-227BS 標準価格200,000円(税別)
会計エキスパートシステムとデータベースを搭載し、機能と操作性を両立させた財務会計ソフト。

CARD PRO-68K
■CZ-226BS 標準価格29,800円(税別)
自由なレイアウト画面で入力できるワープロ機能を装備したカード型リレーショナルデータベース。

CARD PRO-68K用システム手帳リフィル集
■CZ-241BS 標準価格9,800円(税別)

CARD PRO-68K用活用フォーム集
■CZ-242BS 標準価格9,800円(税別)

DATA PRO-68K
■CZ-220BS 標準価格58,000円(税別)
コマンド入力の手間を軽減するヒストリー機能、罫線ドライバー付レポートライター機能、10進31桁の高精度演算。さらにイメージ表示機能を装備したコマンド型リレーショナルデータベースです。

BUSINESS PRO-68K
■CZ-212BS 標準価格68,000円(税別)
スプレッドシート(表計算)、データベース、グラフ作成機能を緊密に一体化させた統合ビジネスツールです。マウス対応のやさしいオペレーション、高度なエディタ機能、豊富な関数群など、初心者からプロまで幅広く使えます。

開発ツール

OS-9/X68000
■CZ-219SS 標準価格29,800円(税別)
X68000のもつグラフィック環境はもちろん、AD PCM音声、FM音源とグラフィックの同時再生といったマルチメディア機能をサポート。OS-9のもつマルチタスク機能、リアルタイム機能を活かした使い易く機能的なOS環境を提供します。また、これまでのデータ資産も活かせます。
※OS-9はマイクロウェア社の登録商標です。

C compiler PRO-68K
■CZ-211LS 標準価格39,800円(税別)

Human68k ver2.0
■CZ-244SS 標準価格9,800円(税別)

THE福袋V2.0
■CZ-224LS 標準価格9,980円(税別)

AI-68K〔Staff LISP/OPS PRO-68K〕
■CZ-234LS 標準価格188,000円(税別)

通信ツール

Communication PRO-68K
■CZ-223CS 標準価格19,800円(税別)
300～19,200BPSまでの通信速度に対応し、各種データベースの漢字端末やパソコン通信に利用できる高機能通信ソフトです。逆スクロール機能や自動実行機能、また豊富な編集機能を装備。

ゲーム

シューティングゲーム
〈ツインビー〉
■CZ-217AS
標準価格7,800円(税別)
©KONAMI. 1988

シューティングゲーム
〈沙羅曼蛇〉
■CZ-218AS
標準価格8,800円(税別)
©KONAMI. 1989

ブロックゲーム
〈アルカノイド〉
■CZ-222AS
標準価格7,800円(税別)
©TAITO CORP. 1987

ドライブゲーム
〈フルスロットル〉
■CZ-231AS
標準価格8,800円(税別)
©TAITO CORP. 1988

スポーツゲーム
〈熱血高校ドッジボール部〉
■CZ-232AS
標準価格7,800円(税別)
©TECHNOS JAPAN CORP. 1988

アクションゲーム
〈パックマニア〉
■CZ-233AS
標準価格7,800円(税別)
©NAMCO

アクションゲーム
〈ニュージーランドストーリー〉
■CZ-230AS
標準価格8,800円(税別)
©TAITO CORP. 1989

スポーツゲーム
〈V'BALL〉
■CZ-246AS
標準価格7,900円(税別)
©TECHNOS JAPAN CORP. 1989

バイクレーシングゲーム
〈スーパーハングオン〉
■CZ-238AS
標準価格8,800円(税別)
©SEGA 1987

ジェットヘリ・シミュレーションゲーム
〈サンダーブレード〉
■CZ-239AS
標準価格9,500円(税別)
©SEGA 1987

アクションゲーム
〈ダウンタウン熱血物語〉
■CZ-254AS
標準価格8,800円(税別)
©TECHNOS JAPAN CORP. 1989

本広告に掲載しております商品および役務の価格には消費税は含まれておりませんので、ご購入の際、消費税額をお支払い下さい。

ジェミニウイング

Gemini Wing™

幾千の流星が降りそそいだ年、世界は蟲に覆われていた。人々は孤立し、街は滅び、植物に埋め尽くされた。蟲たちはさらに勢いを増し、残された僅かな地さえも蝕んでゆく。そして、ついに最高機密指令第307号、コード名ジェミニウイングは発動された……！

MIDI 対応

AD 魅由シリーズ

闇の血族

艶やかなファッション界を襲う奇怪な連続殺人事件。
南米の血に隠された秘密とは?
そして魅由を待ち受ける血族の宿命は?

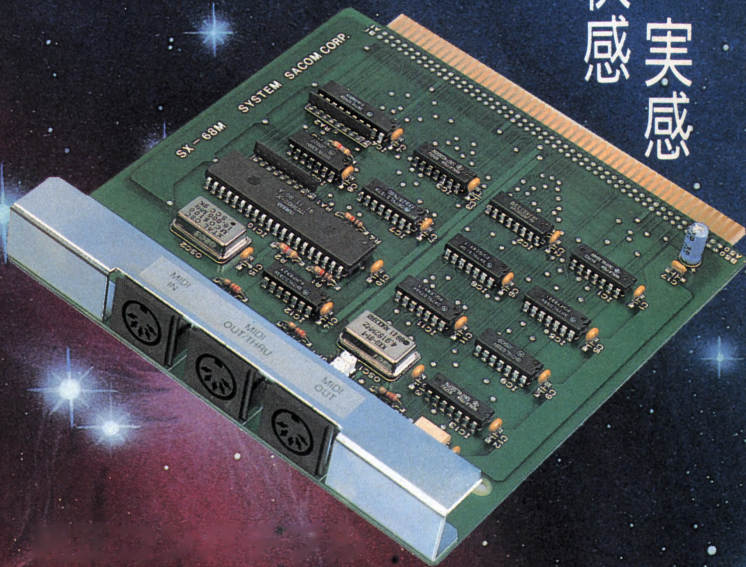
あたし、魅由。
新宿にあるデザイン・スタジオの、新人A・D(アパレル・デザイナー)。……なんだけど、あたしの持ってる妙な「力」みたいなモノ—— 人の心が判っちゃったり、変にカンが良かったり——のせいで、周りからは「名探偵魅由」なんて呼ばれて、よく相談事を持ち込まれたりしている。で、そんなある日、友達のモデルが、突然、殺されてしまった。そして、あたしの親友だった唯も……!
これって……ひょっとして連続殺人事件ってヤツ?!

美少女名探偵 魅由の繰り広げる
ミステリアスアニメーションアドベンチャー第1弾!!

5月中旬発売予定

X68000対応 5"-2HD
標準価格8,800円

・ローランド社
MT-32完全対応

(MIDIインターフェイスボードC-Z-6BMI
又は、SACOM製SX-68Mが必要です。)

SX-68M仕様

品名	MIDIインターフェースボード
規格	MIDI規格 1.0準拠
コントロールLSI	日本楽器(YAMAHA) YM3802
MIDI端子	MIDI OUT 2端子 MIDI IN 1端子
	MIDI OUT 1端子 MIDI THRU 1端子 MIDI IN 1端子
電源	+5V 170mA(本体より供給)
外形寸法	150mm(W)×167mm(D)×23mm(H)
重量	約160g

標準価格 ¥19,800

対応ソフト紹介

■38万キロの虚空

■メタルサイト

株式会社 システム サコム
〒130 東京都墨田区両国4-38-16 両国桜井ビル4F
ハードウェア部 TEL 03(635)5145
ソフトウェア部 TEL 03(635)7609

※標準価格には消費税は含まれておりません。

X68000

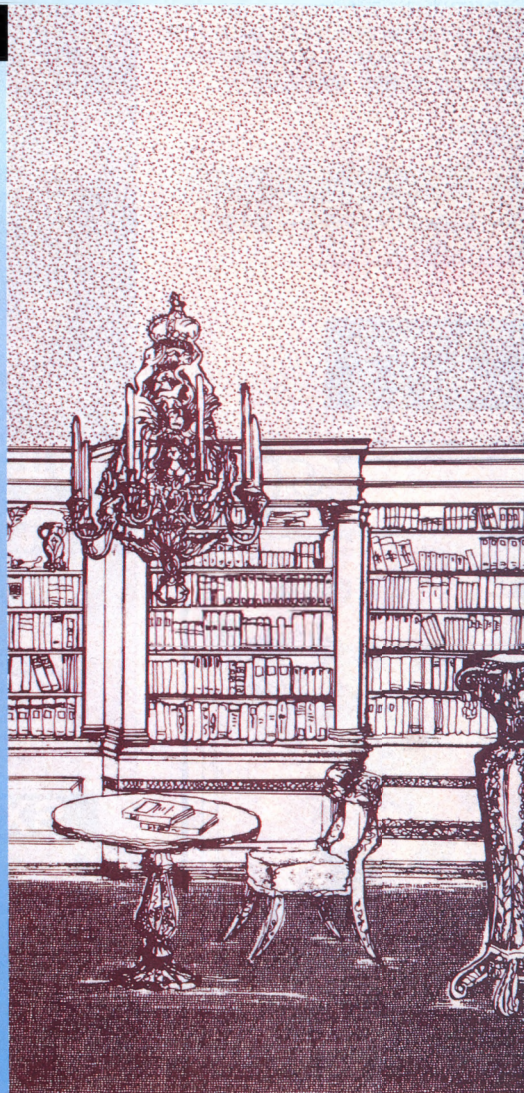
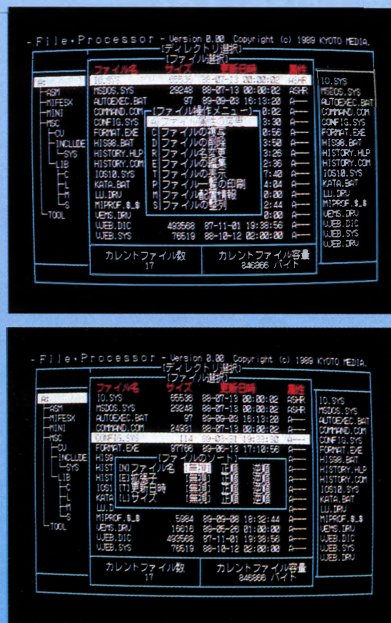
本格的ファイルマネージングソフトウェア

業界の新星、ロゴスシステムが
ユーザーの希望を1つの形にしました。
これは必要だとか便利じゃない、快感だ!

全国有名パソコンショップでお求め下さい。
電話1本での通信販売も受付いたしております。

THE FILE PROFESSORの実力

ディスクのバックアップ、ディスクのエディット、ディスクの初期化、ディスクの比較、ディスクの検査、ディスクの情報、FATのエディット、ファイルの検索、ディレクトリのコピー、ディレクトリの削除、ヴォリュームラベルの設定、ディレクトリの作成、ディレクトリ構造の再読み込み、ディレクトリ構造の印刷、ディレクトリ名の変更、ディレクトリ内容のソート、削除ファイルの復元、ファイル属性の変更、ファイルのコピー/移動、ファイルの削除、ファイルのエディット、ファイルの配置情報、ファイル一覧の印刷、ファイル名の変更、ファイルのソート、ファイル更新日時の変更、ファイルの表示、ファイルの奨行、カレンダー、ハードディスクの直撥エディット、システム情報の表示、コマンドシェル、現在時刻の変更。

メニュー選択方式を実現!!
初心者でも簡単に使える
(画面写真は、98用を開発中のものです。)

The File Professor

ロゴスシステム

このソフトはロゴスシステムのデビュー作です。でも、だからといってなめてもらっちゃぁ困ります。私達は、いろいろなソフトを作りました。そのどれもが他社から発売されていました。出来る事ならば自分達で発売したい!その願いがやっとかないました。

好評発売中!

ロゴスシステム
〒615 京都市右京区西院上今田町17-1 L&Pビル4F
TEL (075) 812-6383　FAX (075) 822-6915

定価28,000円

G68K Version II-PRO 登場！

NEWS!

OH! BUSINESS
●京都市山科区音羽西林町2
サポート室：(075)502-2972
開 発 室：(075)822-4408

▶Exciting graphic tool. ▶High quality. ▶65536 color support.
EXCITING GRAPHIC TOOL FOR X68000
Version 2.0-PRO
G68K II
© 1989 OH! BUSINESS
OH! BUSINESS
PHONE 075-822-4408

発売中！

G68K Version II-PRO

定価：¥22,000

ご 案 内

この度、弊社では発売中のG68Kをバージョンアップ致しましたので、下記のとうりご案内させていただきます。

旧版G68Kは、お求めやすい価格と簡単操作により、入門用ツールとして多くのX68000ユーザーの皆様方よりご好評をいただいております。

今回のバージョンアップでは旧版の簡単操作を継承しつつ、業界でもトップレベルの処理スピードと前作を遥かに上回る、高機能・多機能・高速処理を実現致しました。

旧版G68Kユーザーの皆様方から頂いた多くのご意見を元に、本格的プロ仕様ツールとして大幅バージョンアップ致しました。

サンプルデータもプロのイラストレーターの手に依るコンピュータイラストを収録。また、専用グラフィックデータ集のシリーズ化、発売を近日中に予定いたしております。

高速・高機能・低価格・1MB標準実装のメモリで完全に動作する本格派グラフィックツール。

■前作を大幅に上回る80種類のパレット
- 自由に編集可能
- 模様のついたパレットも作成可能
- HSV方式による色の合成
  色相（色の種類）・彩度（色の濃さ）・明度（色の明るさ）
- 簡単にお望みの色を作り出すための数々の機能を装備
- マスキング塗料・マスク除去塗料を装備
  微妙な修正に威力を発揮
- 2色の混合
- 画面上より自由に色を取り込むスポイト機能
- パレット保存可能
- 画面上より自由にタイルパターンを取り込むタイルパターン用カッターを装備

■32階調の濃淡をもつブラシ
- 自由に形状を変更できるブラシが24種類
- ユーザーが自由に変更・ディスクに保存可能

■大幅に機能アップされたエアブラシ
- ブラシノズル口径、インク噴出速度・濃度を自由に設定

■32階調の濃淡を持つトーンパターン
- 全てのペイントに有効
- 自由に変更・ディスクに保存可能

■強力な編集機能
- 2倍、4倍、8倍に画面を拡大する拡大エディット機能（ルーペ機能）
- 色を調整するカラーコレクタ
- 任意角度の高速画像回転
- 拡大・縮小
- 左右・上下反転
- 切り取りセーブ&ロード
- 自由領域のコピー・移動
- 標準実装のメモリで全画面が編集可能

■製図用具
- マスキング機能
- ペン描画時の直線
- 指定領域のカラー変更
- 円・楕円・ボックス・直線・自由領域
- これらの内部のペイント
- 単色領域ペイント

■文字入力をサポート
- X68000標準24×24ドットキャラクタの表示

■外部機器のサポート
- 豊富な対応周辺機器など ●各種プリンター・イメージスキャナ・カライメージユニット他

■起動直前の画面を保存しながら起動することも可能

■UNDO機能（取り消し処理）
- ペイント等に失敗してもワンステップ前に戻ることが可能

■市販グラフィックツールとのファイルコンバーターが付属
- Z's STAFF-PRO 68Kとのファイル変換が可能

■ノンプロテクト
- ハードディスクへの転送も可能（自由インストール）

■FileはBASICのGL3形式
- BASICより簡単に読み出し可能

▶お問い合わせ・お申し込みは上記電話番号までお願い致します。（上記サポート室迄）

ワンダラーズ フロム イース［イースⅢ］

WANDERERS FROM YS

By Falcom

X68000の為の書き下ろし32曲（新曲６曲）。FM音源とADPCMの絶妙なバランスでくり出す美しいBGMにのせて、高速三重スクロール＋横スクロールで描く遠近感にあふれるグラフィック。また一つ、ゲームソフトの神話が生まれた。　（ジョイスティック対応）

In my time, I've wandered everywhere
Around this world, Hope would always be there

X68000
5′2HD（4枚組）　価格8,700円
好評発売中!!

通信販売　（送料無料）

●代金引換の場合
電話やFAXやハガキで品名・機種名・住所・氏名・年齢・電話番号を明記して申し込み下さい。商品お届け時に商品代金お支払い下さい。

●現金書留の場合
品名・機種名・氏名・電話番号を明記して現金書留で申し込み下さい。

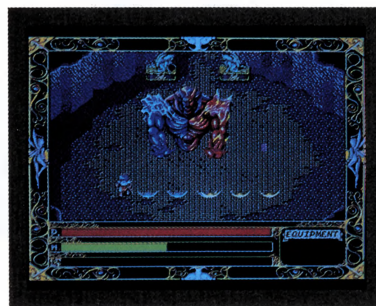
NIHON FALCOM INCORPORATION
〒190 東京都立川市柴崎町2-1-4 ☎0425(27)6501

「X 68000＋VR システム」

90'S RPG最強のコンビネーション!!

©1989 MICRO CABIN

The Art of Visual Stage

4月 新発売!

RPG「Xak」がX68000で新登場!

○68kオリジナルのスーパーグラフィック。
○ボスキャラクター全ニュータイプ。
○攻撃パターンもより複雑に、より多彩になって大迫力!
○BGM68kアレンジバージョン。サンプリング同期FM音源ステレオ対応。
○難易度UP!

X68000 シリーズ
5"2HD 4枚組 ¥8,800(税別)

株式会社 マイクロキャビン
〒510 三重県四日市市安島2-9-12 TEL0593(51)648

FTL

やっぱり凄い！噂を超えた面白さ。

世界中で数々の金字塔を打ち立てたリアルタイムRPG「ダンジョン・マスター」の興奮は本物だった。
3Dグラフィックスに展開される奥の深い迷路、数々のトリック、
パーティーを突然襲って来る不気味なモンスター、組合せと熟練度によって決定される魔法、
それぞれの武器によって異なる攻撃方法、そして何よりもプレイヤーの思考、
行動にリアルタイムで反応する見事なゲーム・システム……
まさにこれこそ本物のリアルタイムRPGだ。

24人の個性あふれるキャラクター

冒険は24人のキャラクターから4人を選ぶことから始まる。それぞれの特性を見極めてパーティーを組むのだ。

魔法は呪文の掛け合わせ。4つの元素がそれぞれ6種なんと計1548の組み合せ。

戦いに必要な魔法はシンボルの組合せで決定。熟練度も加味されてより強力な魔法を編み出せ。

戦いはリアルタイム

持っている武器の特性、パーティーの並び方を瞬時に判断。一瞬の躊躇が命取りになってしまう苛酷な戦闘だ。

豊富なアイテム、武器、防具

プレイヤーの装備は頭から足まで。冒険に必要な水と食物。謎を解明するための鍵や巻物。全てが計算された必要品。

恐怖すら覚える臨場感

音が聞こえる、影がみえる、一歩先に隠された謎やモンスター。リアルタイムRPGのみがもつ緊張感にのめりこむ。

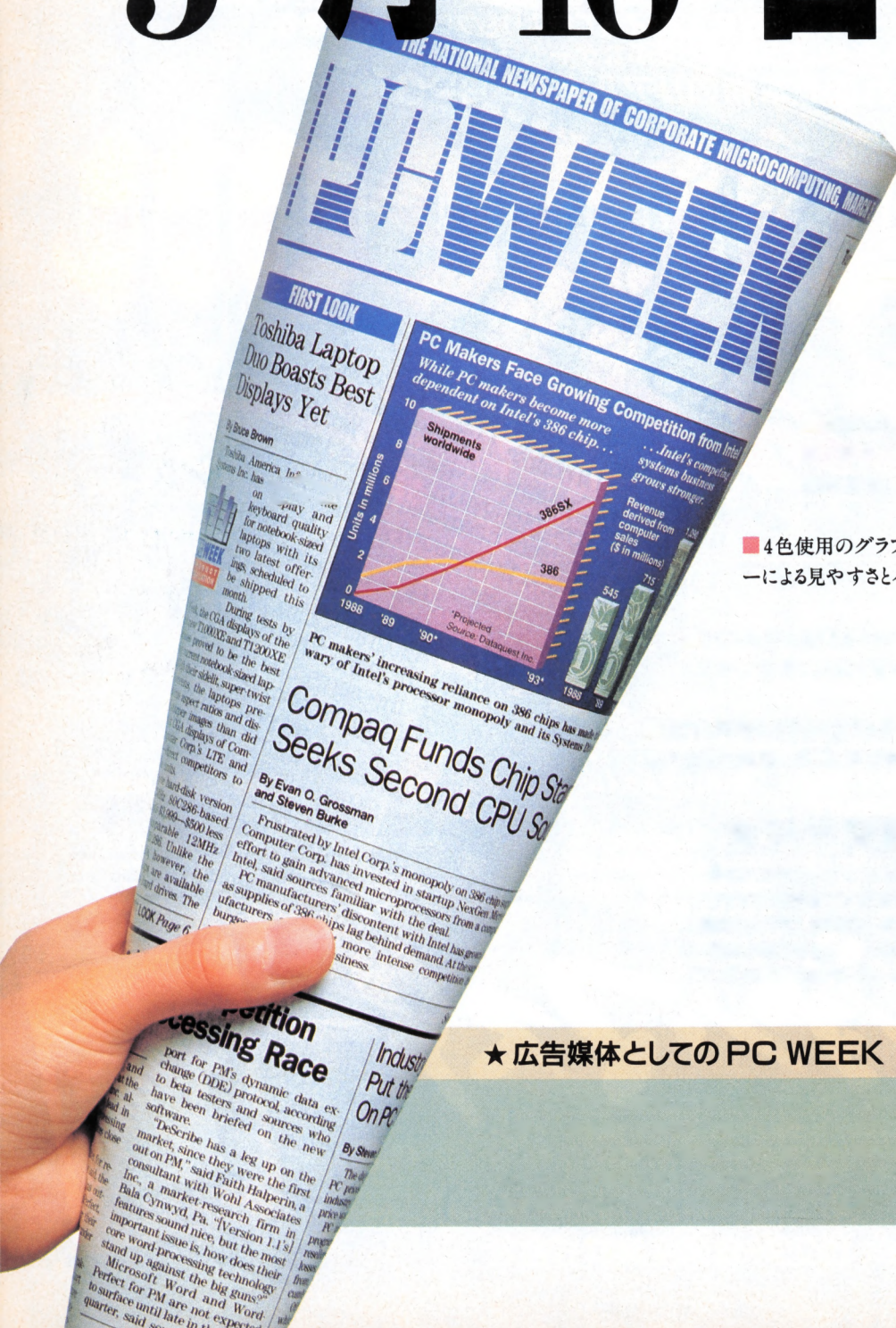
好評発売中 ■X68000 マウス対応 ■PC-9801VM21/11, VX, RX, RS, RA ■PC-98DO ■PC-9801UV21/11, UX, CV, EX, ES 要バス・マウス/アナログRGB対応

各¥9,800（税抜）

「ダンジョン・マスター」オリジナルTシャツ（Mサイズ）
¥1,800（送料込み）にて限定発売！
ご希望のかたは現金書留にて下記通販係までお申し込み下さい。

■発売 ビクター音楽産業株式会社

通信販売 当社の商品をお近くのパソコンショップでお買い求めになれない場合、商品名、機種名、住所、氏名、電話番号を明記のうえ、下記住所まで定価プラス3%消費税分を現金書留にてお申し込み下さい。（送料無料） 〒151 東京都渋谷区千駄ヶ谷2-8-16 ビクター音楽産業株〈通信販売係〉

PC WEEK 日本版 1年間無償購読のご案内

●一定の条件を満たしている企業ユーザーの担当者の方に限り、「PC WEEK」(日本版)の1年間無料購読（送料とも）の特典が提供されます。無料購読を希望される方は、下記の全質問項目の回答を下の葉書にご記入（黒色ボールペンを使用。スタンプなどでの代用は不可）の上、当社当てにご返送下さい。回答番号、数量など全て算用数字でご記入下さい。審査は当社にて厳正に行いますのでご返送後4～6週間ほどお待ちください。

1 お勤め先の主力の事業分野はどれに該当しますか？（1つだけお選びください）

番号を1つだけ→ □

コンピュータ関係以外
1.製造業(非コンピュータ事業)
2.卸・小売り(非コンピュータ事業)
3.機械設計/建築設計/研究開発
4.金融/証券/会計
5.保険/不動産
6.教育
7.官公庁
8.警察/消防/自衛隊
9.病院/診療/法律
10.通信(第一種、第二種)
11.運輸/公益(電力、ガス、水道)
12.コンサルティング(非コンピュータ)
13.その他(非コンピュータ事業/サービス)

コンピュータ関係
14.コンピュータ製造（ハード、ソフト、周辺機器など）
15.コンピュータ流通(卸・販売)
16.コンピュータ販売店
17.VAR/VAD/SI
18 コンピュータ・コンサルタント
19.その他コンピュータ関係事業/サービス

2 あなたの役職名は次のどれになりますか？（1つだけお選びください）

番号を1つだけ→ □

A.会長/社長/理事長/代表者
B.役員/理事/助役
C.事業部長/支配人
D.ネットワーク部門、部課長/部門長
E.コンピュータ技術部門、部課長/部門長
F.MIS/EDP部門、部課長/部門長
G.PC関連部門、部課長/部門長
H.その他コンピュータ関連部門、部課長/部門長
I.コンピュータ関連営業・企画部門、部課長/部門長
J.コンピュータ関連以外の部課長/部門長
K.技術者/科学者
L.PC関連専門職
M.プログラマー/SE/SA
N.コンサルタント
O.その他技術系スタッフ
P.建築設計士/教職者/弁護士/医師
Q.その他専門職
Y.その他職名の個人（正確にご記入下さい）

3 実際に担当されているお仕事は次のどれでしょう？（1つだけお選びください）

番号を1つだけ→ □

A.データ/通信
B.システム開発/設計
C.広報/宣伝/サービス
D.MIS/情報処理
E.研究開発
F.経理/財務
G.管理監督
H.製図/デザイン
I.教育/研修
J.オペレーション/品質管理
K.購買
L.営業/マーケティング
M.コンサルティング
Z.その他（正確にご記入下さい）

4 次の商品の購入決定に際して何を重視しますか（該当する全てをチェックして下さい）

	必要機能	ブランドの評価	メーカーの仕様説明	納入業者の仕様説明	再販契約など	ユーザーの評価	その他
パソコン	10.□	11.□	12.□	13.□	14.□	15.□	16.□
ソフトウェア	20.□	21.□	22.□	23.□	24.□	25.□	26.□
周辺機器	30.□	31.□	32.□	33.□	34.□	35.□	36.□
ネットワーク機器	40.□	41.□	42.□	43.□	44.□	45.□	46.□

5 過去1年間に何台のPC購入にかかわりましたか（関与していない場合は0とご記入下さい）

あなたの会社(団体)または個人が使うため A. 台
他の企業に対する販売または再販の目的で B. 台
コンサルタントの結果としてお客様が購入 C. 台

6 何台のPCについて、ソフトウェア、周辺機器、データ・ネットワーク機器の購入にかかわりましたか（関与していない場合は0とご記入下さい）

台

7 あなたの会社の事業形態が、販売、再販、コンサルティングに該当する場合、過去1年間に何台ぐらいのワークステーションを売られ(購買指導され)ましたか

台

▶こちらのハガキに
アンケートのお答えを記入して
今すぐお申込下さい

郵便はがき

料金受取人払
麹町局承認
2089

差出有効期間
1991年11月30日まで

102-00

東京都千代田区九段南3-3-6
日本ソフトバンク出版事業部
営業部 PC WEEK 日本版 行

「PC WEEK」(日本版)の1年間無償購読を
□希望します □希望しません

送付先会社（団体）所在地：〒
企業（団体）名：
部署名：
ご役職名：
ご氏名：
電話番号： － －

米国PC MAGAZINE／パソコン・マガジン第1回共同セミナー

Catch PC Waves in 90s

90年代のパソコンはこうなる。

米国PC MAGA[illegible]Eのエグゼクティブスタッフを
[illegible]おくる日米共[illegible]ミナー。
OS、ハードウェア[illegible]フト[illegible]ームなどの標準化[illegible]題か[illegible]
ネットワーク、流通、そ[illegible]て米[illegible]場における成功[illegible]といっ
年代[illegible]イバルに欠くこと[illegible]きな
業[illegible]ンドのすべてを[illegible]しま

【日時】＝平成2年5月16日(水)午後1時より6時まで
【場所】＝ホテルニューオータニ はぎの間(東京・紀尾井町)
【参加費用】＝1万円(消費税込み)

Eric Hippeau

Bill Machrone

William F. Zachmann

John C.Dvorak

ジョン・C・デュボラック。コンピュータ業界の裏情報をふんだんに盛り込んだ軽快なタッチと鋭い批評で知られる有名コラムニスト。現在は、PC MAGAZINE、PC Computing、MacUserにもレギュラーで執筆中。

90年代のコンピューティングプラットフォーム
米国市場における成功の鍵(企業ユーザーの購買トレンド)
アプリケーション開発の今後
90年代のマーケティング
パネルディスカッション(質疑応答)

◉出席者(予定)
エリック・ヒッポー(米国PC MAGAZINE発行人)
ビル・マクローン(米国PC MAGAZINE編集長兼出版ディレクター)
ウィリアム・F・ザックマン(米国PC MAGAZINEコラムニスト)
ジョン・C・デュボラック(米国PC MAGAZINEコラムニスト)
中村明彦(パソコン・マガジン編集長)
富田倫生(パソコン・マガジンコラムニスト)

◉特別ゲスト(予定)
ジム・マンジ(米国ロータスディベロップメント社長)
溝口哲也(東芝パソコンワークステーション事業部長)
古川 享(マイクロソフト社長)
脇 英世(東京電機大学工学部電気通信工学科助教授)

(すべてのセッションで日英語の同時通訳サービスを行ないます)

※出席者および特別ゲストは変更になる場合もございます。

◉参加希望のかたは官製ハガキに住所、氏名、参加人数、会社名、連絡先電話番号を明記の上、5月7日(月)必着で下記までお送りください。先着順に申込書をお送りいたします。

郵便番号102
東京都千代田区九段南3-3-6 日本生命麹町ビル1階
株式会社日本ソフトバンク出版事業部
第1回PC MAGAZINEセミナー係

◉お問合せ先 広告営業部 03-230-7672

超低金利クレジットOK

注目!!

夏のボーナス一括払い
手数料(金利)無料
(6月末・7月末どちらかご指定下さい)

また また 秋葉原で おなじみの

4/15〜5/15

●お近くの方はお
●本体単品で特
●ビジネスソフト定

CYBER STICK
●CZ-8NJ2
(定価￥23,800)
超特価!!
▶価格はTEL下さい

X68000シリーズ専用 特価￥16,480
MIDIインターフェースボード
SX-68M(サコム)
(純生コンパチ)定価￥19,800

X-1ターボZIII 特別ご提供品!! 台数限定
●CZ-888C＋CZ-860D＋M-2HD(10枚)
定価￥269,600▶特価￥164,800
・ジョイカード
・ゲーム3種
・パソコンラック(A)3段
プレゼント中
(ボーナス併用も有りますTEL下さい)
送料消費税込み!!

12回	14,300	24回	7,500	36回	5,100	48回	4,000	60回	3,300

ジョイスティック　送料￥500
●X-1PRO
定価￥9,500▶特価￥7,800
●ASCII STICK
定価￥6,800▶特価￥5,500

P&A超低金利クレジットをご利用ください!!

X68000EXPERTII & EXPERTII-HD (送料消費税込み)

NEW

EXPERTII & PRIIセット
でお買い上げの方に、
●ディスケット 10枚
●ゲーム 2種
●ジョイカード
プレゼント中!!

EXPERTII

Ⓐセット:CZ-603C＋CZ-604D ……定価￥432,800▶特価(価格はお電話下さい。)

12回	29,700	24回	15,600	36回	10,700	48回	8,300	60回	6,900

Ⓑセット:CZ-603C＋CZ-605D ……定価￥453,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:CZ-603C＋CZ-613D ……定価￥473,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓓセット:CZ-603C＋CU-21HD ……定価￥486,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

EXPERTII-HD

Ⓐセット:CZ-613C＋CZ-604D ……定価￥542,800▶特価(価格はお電話下さい。)

12回	37,000	24回	19,400	36回	13,300	48回	10,400	60回	8,600

Ⓑセット:CZ-613C＋CZ-605D ……定価￥563,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:CZ-613C＋CZ-613D ……定価￥583,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓓセット:CZ-613C＋CU-21HD ……定価￥596,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

X68000PROII & PROII-HD (送料消費税込み)

NEW

EXPERTII & PRIIセット
でお買い上げの方に、
●ディスケット 10枚
●ゲーム 2種
●ジョイカード
プレゼント中!!

PROII

Ⓐセット:CZ-653C＋CZ-604D ……定価￥379,800▶特価(価格はお電話下さい。)

12回	26,100	24回	13,700	36回	9,400	48回	7,300	60回	6,100

Ⓑセット:CZ-653C＋CZ-605D ……定価￥400,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:CZ-653C＋CZ-613D ……定価￥420,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓓセット:CZ-653C＋CU-21HD ……定価￥433,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

PROII-HD

Ⓐセット:CZ-663C＋CZ-604D ……定価￥489,800▶特価(価格はお電話下さい。)

12回	33,900	24回	17,700	36回	12,200	48回	9,500	60回	7,900

Ⓑセット:CZ-663C＋CZ-605D ……定価￥510,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓒセット:CZ-663C＋CZ-613D ……定価￥530,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

Ⓓセット:CZ-663C＋CU-21HD ……定価￥543,000▶特価(価格はお電話下さい。)

12回	?	24回	?	36回	?	48回	?	60回	?

X68000シリーズ 〜P&Aスペシャルセット＝限定誌上販売!!

台数限定　送料、消費税込み　●ディスケット10枚●ゲーム2種●ジョイカードプレゼント中

EXPERT

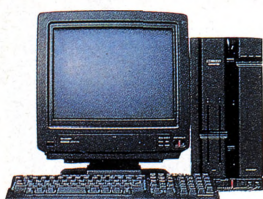
●CZ-602C＋CZ-612D ……定価￥475,800▶特価¥305,000
●CZ-602C＋CZ-604D ……定価￥450,800▶特価¥299,000
●CZ-602C＋CZ-605D ……定価￥471,000▶特価¥319,000
●CZ-602C＋CZ-613D ……定価￥491,000▶特価¥336,000
●CZ-602C＋CU-21HD ……定価￥504,000▶特価¥338,000

EXPERT-HD

●CZ-612C＋CZ-612D ……定価￥585,800▶特価¥374,000
●CZ-612C＋CZ-604D ……定価￥560,800▶特価¥368,000
●CZ-612C＋CZ-605D ……定価￥581,000▶特価¥388,000
●CZ-612C＋CZ-613D ……定価￥601,000▶特価¥405,000
●CZ-612C＋CU-21HD ……定価￥614,000▶特価¥407,000

PRO-HD

●CZ-662C＋CZ-612D ……定価￥527,800▶特価¥337,000
●CZ-662C＋CZ-604D ……定価￥502,800▶特価¥331,000
●CZ-662C＋CZ-605D ……定価￥523,000▶特価¥351,000
●CZ-662C＋CZ-613D ……定価￥543,000▶特価¥368,000
●CZ-662C＋CU-21HD ……定価￥556,000▶特価¥370,000

●本広告の掲載の商品の価格については、消費税は含まれておりません。●営業時間＝平日AM10:00〜PM7:00、日祭AM10:00〜PM6:00

回〜60回払いまでOK!!

★頭金なし! ★即日発送

P&AがズバリZ超特価セールでご奉仕!!

立寄り下さい。専門係員が説明いたします。
価で受付します。詳しくは電話にてお問合せ下さい。
価の20%引きOK! TELください。

全国通販

超特価でクレジットが組める!!

X68000用ソフトコーナー(送料1ヶ〜5ヶまで¥500)

品名	定価	特価
Z's STAFF PRO68K Ver2.0(ツァイト)	定価¥ 58,000	特価¥ 40,000
C-TRACE68(キャスト)	定価¥ 68,000	特価¥ 50,000
サイクロン エキスプレス(アンス・コンサルタンツ)	定価¥ 78,000	特価¥ 57,000
Z's TRIPHONY デジタル クラフト(ツァイト)	定価¥ 39,800	特価¥ 29,300
テラッツォ(ハミングバード)	定価¥ 19,800	特価¥ 15,800
G-68K(OH! BISINESS)	定価¥ 14,800	特価¥ 11,400
KAMIKAZE(サムシング・グッド)	定価¥ 68,800	特価¥ 46,000
EW&EI(イースト)	定価¥ 38,800	特価¥ 28,800
C&Professional Pack(マイクロウェアジャパン)	定価¥ 58,800	特価¥ 43,000
Final Ver3.2(エーエスピー)	定価¥ 38,000	特価¥ 30,000
DATA PRO68K CZ220BS	定価¥ 58,000	P&A特価 TEL下さい。!
CARD PRO68K CZ226BS	定価¥ 29,800	
C compiler PRO68K CZ211LS	定価¥ 39,800	特価¥ 32,000
OS-9/X68000 CZ219SS	定価¥ 29,800	P&A特価 TEL下さい。
AI-68K CZ234LS	定価¥188,000	特価¥143,000
THE福袋V2.0 CZ224LS	定価¥ 9,980	特価¥ 18,000
SOUND PRO68K	定価¥ 15,800	特価¥ 12,500
MUSIC PRO68K CZ213MS	定価¥ 15,800	P&A特価 TEL下さい。
Sampling PRO68K CZ215MS	定価¥ 17,800	特価¥ 14,000
MUSIC-studio PRO68K 237MS	定価¥ 15,800	P&A特価 TEL下さい。
MUSIC-PRO68K〔MIDI〕 247MS	定価¥ 18,800	特価¥ 22,000
New-print Shop 221HS	定価¥ 19,800	P&A特価 TEL下さい。!
Communication 223CS	定価¥ 19,800	

X68000用ハードディスク (送料¥1,000)

アイテム
- HXD-040(40MB/23ms)……定価¥118,000▶特価¥ 88,000
- HXD-042(増設用)……定価¥128,000▶特価¥ 95,000

アイテック
- ITX-640(40MB/28ms)……定価¥158,000▶特価¥ 98,500
- ITX-680(80MB/20ms)……定価¥198,000▶特価¥127,000

プリンター(ケーブル・用紙付)限定5台 新品(送料¥1,000)

- CZ-8PC3(カラー漢字24ドット熱転写プリンター)
  定価¥65,800……特価¥39,800
- CZ-8PK8(24ピン漢字プリンター136桁)
  定価¥152,000……特価¥75,800
- CZ-8PC4 P&A特選!!
  定価¥99,800……特価¥73,800

周辺機器コーナー(送料¥1,000)

品名	定価	特価
Ⓐ CZ-8NSI	定価¥188,000	特価¥145,000
Ⓑ CZ-6VTI	定価¥ 69,800	特価¥ 54,000
Ⓒ CZ-6TU	定価¥ 33,100	特価TEL下さい。
Ⓓ BF-68PRO	定価¥ 19,800	特価¥ 15,500
Ⓔ CZ-6BEI	定価¥ 35,000	特価¥ 26,500
Ⓕ CZ-6BEIA	定価¥ 38,000	特価TEL下さい。
Ⓖ CZ-6BE2	定価¥ 79,800	特価TEL下さい。
Ⓗ CZ-6BE4	定価¥138,000	特価¥107,000
Ⓘ CZ-6BFI	定価¥ 49,800	特価TEL下さい。
Ⓙ CZ-6BPI	定価¥ 79,800	特価¥ 61,000
Ⓚ CZ-6BMI	定価¥ 26,800	特価TEL下さい。
Ⓛ CZ-6EBI	定価¥ 88,000	特価TEL下さい。
Ⓜ AN-S100	定価¥ 36,600	特価¥ 28,500
Ⓝ CZ-6SDI	定価¥ 44,800	特価¥ 35,000
Ⓞ CZ-8PC3	定価¥ 65,800	P&A超特価 TEL下さい。
Ⓟ CZ-8PC4	定価¥ 99,800	
Ⓠ CZ-8PG1	定価¥130,000	
Ⓡ CZ-8PG2	定価¥160,000	
Ⓢ CZ-8PK10	定価¥ 97,800	
Ⓣ CZ-6PVI	定価¥198,000	特価¥153,000
Ⓤ IO-735X	定価¥248,000	特価TEL下さい。
Ⓥ CZ-8BSI	定価¥ 23,800	特価¥ 19,000

モデムコーナー (送料¥1,000)

品名	定価	特価
Ⓐ MD-24FS5(オムロン)	定価¥ 49,800	特価¥ 34,800
Ⓑ MD-24FS7(オムロン)	定価¥ 64,800	特価¥ 45,000
Ⓒ コムスター2424/4(NEC)	定価¥ 38,800	特価¥ 28,000
Ⓓ コムスター2424/5(NEC)	定価¥ 44,800	特価¥ 32,000

P&A 特選パソコンラック (送料無料)移動自由(キャスター付)

中古パソコン　送料¥2,000

- X-68000セット……▶¥210,000
- X-68000ACEセット……▶¥240,000
- X:1ターボZセット……▶¥100,000
- X-1G/30セット……▶¥ 39,000
- CZ-822C……▶¥ 15,000
- CZ-830C……▶¥ 25,000
- CZ-856C……▶¥45,000
- CZ-870C……▶¥55,000
- CZ-881C……▶¥65,000
- CZ-820D……▶¥10,000
- CU-14GB……▶¥ 5,000
- CU-14BD……▶¥25,000
- CU-14AG2……▶¥30,000
- CU-14H2……▶¥30,000
- CZ-8PC2……▶¥25,000
- CZ-8PK6……▶¥32,000

中古パソコンはP&Aにおまかせ!!

その場で高価現金買取り・高価下取りOK!!

■まずはお電話下さい。
03-651-1884
FAX:03-651-0141

■下取り・買取りでお急ぎの方、直接当社に来店、または、宅急便にてお送り下さい。

●下取りの場合……価格は常に変動していますので査定額をお電話で確認して下さい。(差額は、P&A超低金利クレジットをご利用下さい。)

●買取りの場合……現品が着き次第、2日以内に買取り金額を連絡し、振込み、又は書留でお送り致します。

●近郊の方は、P&A本店まで、直接お持ち下さい。
即金にて、¥1,000,000までお支払い致します。

通信販売お申し込みのご案内

〔現金一括でお申し込みの方〕
●商品名およびお客様の住所・氏名・電話番号をご記入の上、代金を当社まで、現金書留でお送りください。(プリンター・フロッピーの場合、本体使用機種名を明記のこと)

〔銀行振込でお申し込みの方〕
●銀行振込ご希望の方は必ずお振込みの前にお電話にてお客様のご住所・お名前・商品名等をお知らせください。
(電信扱いでお振込み下さい。)

〔振込先〕住友銀行 新小岩支店
当No.263914 ㈱ピー・アンド・エー

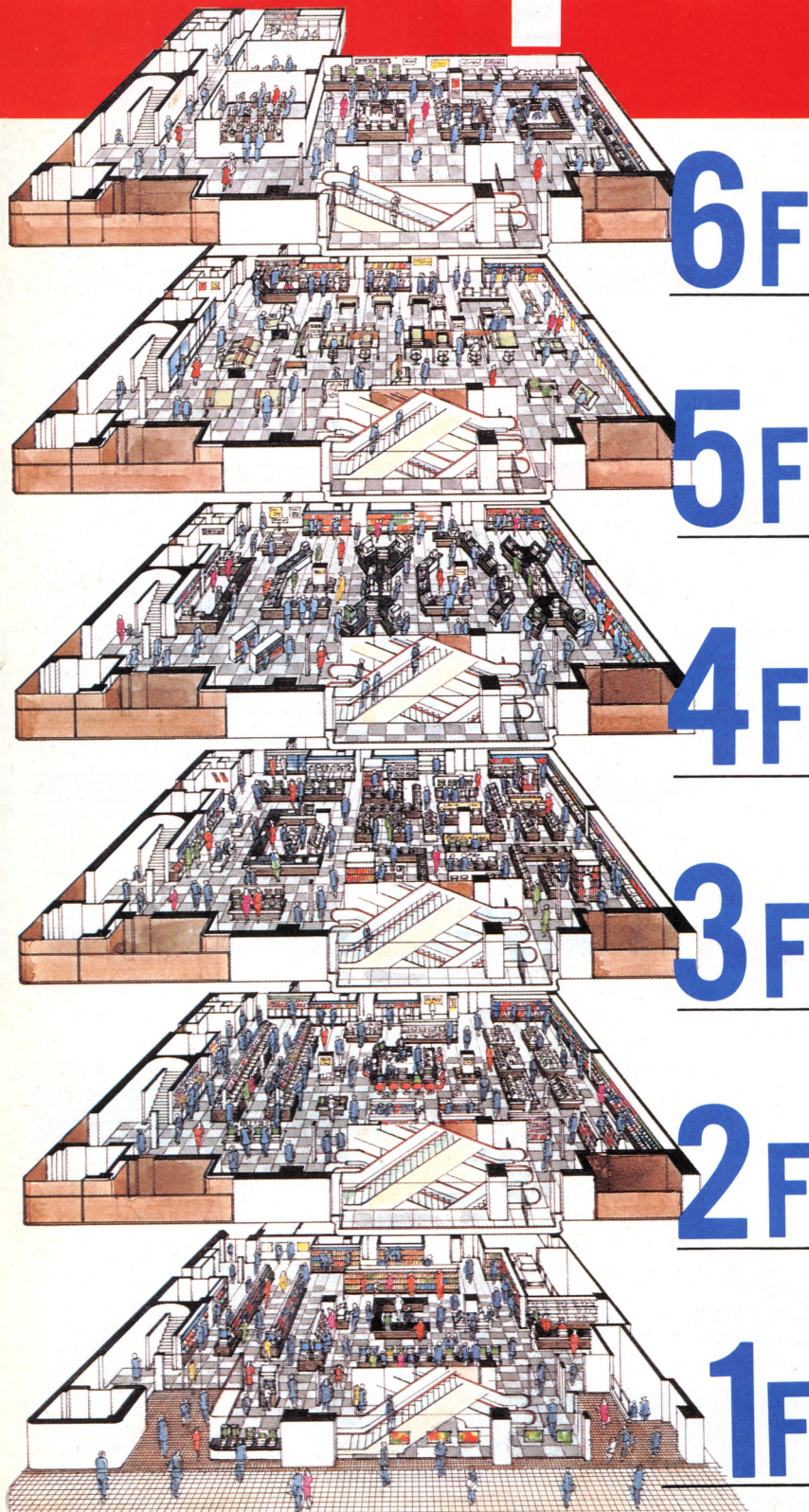
〔クレジットでお申し込みの方〕
●電話にてお申し込みください。クレジット申し込み用紙をお送りいたしますので、ご記入の上、当社までお送りください。
●現金特別価格でクレジットが利用できます。残金のみに金利がかかります。
●1回〜60回払いまで出来ます。但し、1回のお支払い額は3,000円以上。

アフターサービス万全
全商品保証付。専門の担当者がお客様の立場で対応します。
初期不良、輸送トラブルetc.
万が一初期不良、輸送トラブルが発生しました際には、即交換させていただきます。

●定休日/毎週水曜日=第3水曜・木曜は連休とさせていただきます(祭日の場合は翌日になります)

超低金利クレジット率

回数	1	3	6	10	12	15	18	24	36	48	60
利率(%)	1.5	2.0	3.0	4.5	4.5	7.5	9.0	9.5	13	17	22

●現金書留及び銀行振込でお申し込みの方は、上記商品の料金に3%加算の上でお申し込み下さい。詳しくは、お電話でお問い合せ下さい。

THE 日本一のコンピュータ館です
COMPUTER館
秋葉原に、
6F
5F
イベントフロア
セミナールーム・催事場
毎日、毎日が楽しい催物や、役立つセミナーでいっぱい！ザ・コンピュータ館は目が離せません。
4F
アップルコーナー
ソフトから周辺機器まで豊富な品揃え
Laox NET・パソコン通信用機器及び体験コーナー
3F
2F
1F
ホビー&BOOKフロア
書籍・ホビーパソコン/周辺機器・ゲームソフト・インフォメーションカウンター・THE COMPUTER今月号特集コーナー
※フロア名は仮称です。(一部変更になる場合もございますのでご了承ください
The COMPUTER館は、新製品のハードや周辺機器及びソフトをいち早く展示し、体験できます。

4/29(日) OPEN!

日本最大規模の
ショップが誕生します。

THE COMPUTER館はマルチメディア

最新のソフトウェア、ハードウェアを一同に会した情報発信基地です。 だから豊富な品揃えと魅力ある展示で日本一をめざします。

この店は まるでコンピュータのメリーゴーランド

常に変わり続け、発展するシステムや続々と発表される新製品などに対応して、売場も常に変身します。そして、いつもどこかの売場で、コーナーでエキサイティングなイベントやフェアが繰りひろげられている楽しいお店です。

THE COMPUTER館は システムで対応します

■CAD/CAM/CG
●オペレーション教育 ●インストラクター派遣 ●部品登録 ●アプリケーション開発 ●デモンストレーションなど

■Net Work
●異機種接続 ●システム提案 ●コンサルティング ●LAN OS販売

■VAR
●OA化コンサルティング ●システム設計 ●業務診断 ●受託開発 ●セミカスタマイズ ●アプリケーションサポート ●OA導入指導

パソコンライフを拡げる30,000点のソフト!!

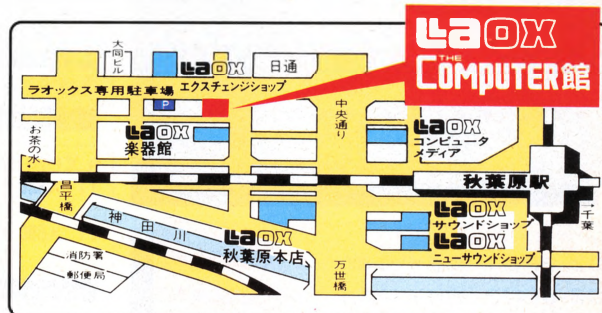
ビジネスはもちろん、CAD、CAM、グラフィックスにホビーを加え、30,000点のソフトを取り揃え、まさに、ないソフトはない! そんなお店です。

●自信がなければ、できないお約束!

ご予約・ご注文のソフトが、お約束の日時までに揃えられなかった場合は、オリジナル・オレンジCARDを差し上げます。

日時 4月29日(日)・30日(振)・5月1日(火) 3日間限り

乗って、乗って、乗りまくれ! グレードアップしたX68000新シリーズによる楽しさいっぱいの"MIDI"体験。パソコンミュージシャンの奏者による、本格的"MIDI"演奏と"トーク"でX68000"MIDI"システムのすべてをコミュニケートします。皆様お誘い合わせのうえ、ぜひ、ご来店ください。

その他、楽しく役立つイベントやソフトのデモ実演など最先端の情報がいっぱいです。

ザ・コンピュータ館

TEL/03-5256-3111

〒101 東京都千代田区外神田1-7-6(秋葉原駅徒歩4分) P駐車場完備

NEXT X68000

つ・い・に!! 新発売!

通信販売のお申し込みは受注専用

フリーダイヤル 0120-377-999

商品についてのお問い合せは各店又は

通信販売部 03(251)9911

★予約受付中!

SUPER HD CZ-623C ¥498,000

更にソフトウェア パワーアップして新登場!!

EXPERT II シリーズ
- CZ-603C……………定価 ¥338,000
- CZ-613C……………定価 ¥448,000

PRO II シリーズ
- CZ-653C……………定価 ¥285,000
- CZ-663C……………定価 ¥395,000

♪ Let's Music ♬

MIDIプレイヤーAセット
- CM-32L……………¥69,000
- SX-68M……………¥19,800
- Musicstudio Mu-1……………¥19,800

合計定価¥108,600
ツクモ特価 ¥91,800 (消費税別途¥2,754)
クレジット例(税込)月々¥5,780×18回払

MIDIプレイヤーBセット
- CM-64……………¥129,000
- SX-68M……………¥19,800
- Musicstudio Mu-1……………¥19,800

合計定価¥168,600
ツクモ特価 ¥144,000 (消費税別途¥4,320)
クレジット例(税込)月々¥7,050×24回払

★Musicstudio PRO-68K V1.1又は、MusicPRO68K(MIDI)のソフトの場合には¥8,000プラスになります。

★旧製品は、更に安く提供中、お電話下さい

台数限定

- CZ-652C (PRO)……………¥298,000
- CZ-662C (PRO HD)……………¥408,000
- CZ-602C (EXPERT)……………¥356,000
- CZ-612C (EXPERT HD)……………¥466,000

★X68000用メモリーボード

IOデータ
- PIO-6BE1-A 定価¥25,000 特価¥21,500
- PIO-6BE2-2M 定価¥50,000 特価¥42,500
- PIO-6BE4-4M 定価¥88,000 特価¥74,500

※2MBと4MBは全てシリーズ対応拡張スロット用。

★Software tools

GRAPHIC TOOLS
- ●マジックパレット……………特価¥16,830
- ●Z's STAFF PRO-68K……特価¥49,300
- ●サイクロンExpress……………特価¥66,300
- ●デジタルクラフト……………特価¥33,800

電子手帳ソフト
- ●CYBERNOTE PRO-68K……定価¥19,800
- ●Stationery PRO-68K……定価¥14,800

通信ソフト
- ●通信ソフト た〜みのる2……………特価¥15,000

★TSUKUMO NET

新規会員募集!! この度、X68000PROのホストシステムへ移行し、3回線までサポートしました。

入会希望の方は7号店荒井まで!

回線番号 ☎03(253)2464
ゲストOK!

★モデム

一流メーカー 2400bps(クラス4)
定価¥38,800 特価¥29,800

アイワ PV-A24MNP5
定価¥54,800 特価~~¥39,900~~

オムロン MD-24FS5 (2400ボー/クラス5)
定価¥49,800 特価¥39,800

★プリンター

CZ-8PG1	定価¥130,000	ツクモ特価販売中!
CZ-8PG2	定価¥160,000	ツクモ特価販売中!
CZ-8PC3	定価¥65,800	ツクモ特価販売中!
CZ-8PC4	定価¥99,800	ツクモ特価販売中!
IO-735X	定価¥248,000	ツクモ特価販売中!

★特価はお電話にてお問い合せ下さい!

★X68000用ハードディスク アイテック

IT X640 定価¥158,000
ツクモ特価¥128,000

IT X680 定価¥198,000
ツクモ特価¥158,000

(カラー:ブラックとグレー)

★ポケコン&電子手帳

- PC-E500PJ 特価¥24,800 限定品
- PA-8600 特価¥24,800
- PA-7500 特価¥17,800

ツクモグローバルカード

入/会/者/募/集
国内・外で活躍

ツクモグローバルカードはジャックスVISA、セントラルマスターとの提携カードです。ツクモのお買い物がらくらくできるうえに国内はもとより海外での分割ショッピングもOK/18才以上の方なら学生でもOK/

18才以上なら学生でもOK!

お申し込みは
(03)251-9898
又は各店店頭で…

秋葉原各店

営 AM10:15〜PM7:00 休 5/3を除く毎週木曜日

★表示価格には消費税は含まれておりません。

ツクモは「スーパーX PRO SHOP」です。

PRO STAFF ツクモ

九十九電機株
〒101-91 東京都千代田区神田郵便局私書箱135号

7号店 荒井

N.C店 福地

★商品のご注文は在庫確認の上お願いします。

ツクモ7号店 ☎03-253-4199(担当/荒井)

便利で安心な通信販売
通信販売部☎03-251-9911

- ■ツクモ5号店 ☎03-251-0531(担当/川名)
- ■ニューセンター店 ☎03-251-0987(担当/福地)
- ■名古屋1号店 ☎052-263-1655(担当/吉高)
- ■名古屋2号店 ☎052-251-3399(担当/横山)
- ■ツクモ札幌 ☎011-241-2299(担当/村井)

カード払い	全国代金引き換え配達	クレジット払い	現金書留払い	銀行振込払い	各種リース払い
通信販売での御利用カード、ツクモグローバルカード、VIPカード、セントラル、ジャックス ※御本人様より電話で通信販売部へお申し込み下さい。	お申し込みは☎03-251-9911へお電話1本! 配達日の指定もできます	月々¥3,000以上の均等払いも頭金なし、夏・冬ボーナス2回払いも受付中	〒101-91 東京都千代田区神田郵便局私書箱135号 九十九電機株通信販売部 Oh!X係	事前に☎でお届け先をご連絡下さい。 富士銀行 神田支店(普)No.894047 九十九電機株	くわしくは各店にお問い合せ下さい ケースに合わせてご相談にのらせて頂きます

★クレジット例は3/31現在の金利手数料で計算されておりますので金利が変わった場合、クレジットの金額が変わりますが御了承下さい。

予告

愛されて1000B年，いま感謝を込めて特別企画

Oh!X6月号を買うと，

あの，X68000用SUPER-HD?がもれなく！

［収録内容］

押し寄せるプログラム
言語関係，数値演算ドライバ関係，音楽ドライバ関係，
ゲーム関係，3Dグラフィック関係，全機種共通システム関係，
その他ユーティリティ関係。
X1turbo用のプログラムは掲載されるのか？
悪魔のツールとはいったい？

［注意事項］

Oh!X 1990年6月号は諸々の事情により特別定価（税込）780円となります。要するにオマケがつきます。検討の結果，メディアは5インチ2HD(Human68kフォーマット）と決定しました。5インチ2D，3.5インチ2DD/2HD，3インチ2D，クイックディスク，およびカセットテープユーザーの方はあらかじめご了承ください。

容量的な問題からディスクには実行ファイル優先で収録されます。ソースファイルは付属しない場合が考えられますのであらかじめご了承ください。

また，ひょっとしたらあなたが苦労して打ち込んだあのプログラムや，多額の電話代を使ってダウンロードしたあのプログラムがポンと収録されている可能性がありますのであらかじめご了承ください。

ディスクには大量のプログラムが詰め込まれる予定ですので，6月号の付録ディスクを立ち上げる場合にはフォーマット済みの2HDディスクが2枚程度必要になる可能性があります。あらかじめご了承ください。

一応，X68000用となっていますが，それ以外の機種用のプログラムが収録されている可能性もあります。

なお，フロッピーはSUPER-HDでお馴染みの富士写真フイルムの協力によるオリジナルディスクです。

新機種発表！

X68000 SUPER/EXPERT II/PRO II

80MバイトHD搭載モデルを加えた新ラインアップ
シリーズ全機種にSX-WINDOWをサポート
光磁気ディスクなど周辺機器も充実

新ラインアップと充実の環境

すでに一部の新聞や雑誌などでご存じの方も大勢おられると思うが，今年もX68000に新しい仲間が登場した。今回発表されたのは，EXPERT/PROシリーズを継承するX68000EXPERT IIシリーズおよびPRO IIシリーズ，そして新たにラインアップに加わったX68000SUPER-HDである。

これまでと同様，EXPERT IIとPRO IIにはそれぞれ40Mバイトのハードディスクが内蔵されたEXPERT II-HDとPRO II-HDがあり，それぞれにブラックとオフィスグレーの2色が用意されている。価格はEXPERT IIが338,000円で，PRO IIが285,000円，ハードディスクタイプのEXPERT II-HDおよびPRO II-HDはそれぞれちょうど11万円高い448,000円と395,000円だ。

また，SUPER-HDはSCSIを採用し80Mバイトのハードディスクを内蔵した最上級機種である。価格は498,000円となっている。

そして今回の最大の目玉は，シリーズ全機種に搭載されたウィンドウシステム「SX-WINDOW」だ。SX-WINDOWはユーザーフレンドリーなビジュアルインタフェイスとして優れた操作環境を実現するとともに，ウィンドウ上のアプリケーションを容易に開発するためのツールボックスを備えた本格的なウィンドウシステムとなっている。

また，全機種ともBIOSが高速化されるなど細かい改良がなされている。

このほか，周辺機器も充実。ディスプレイテレビ2機種に加え，ついに登場した光磁気ディスクユニット，SCSIボード，そしてビデオ出力ボードも発表された。さらに，X68000ACE/PRO/PRO II用の1Mバイト増設RAMボードが従来の38,000円から28,000円に値下げされるなど，うれしいニュースがいっぱいだ。

チタンブラックのSUPER-HD

SUPER-HD。X68000のなかで最高級の位置づけにあるのがこのX68000SUPER-HD。このモデルだけは従来のモデルと色が違い，なんとチタンブラックというカラーを採用している。写真ではわかりにくいが，オーディオ機器などではわりと流行の色だ。その格調高い色合いは外見にも相応のこだわりをもつ潜在ユーザーを掘り起こすことになるかもしれない。近くにパソコンショップがある人はぜひとも現物を一度見ておきたい。

さて，X68000SUPER-HDの最大のポイントは，80Mバイトの3.5インチハードディスクを内蔵しているということだ。これに伴い，SUPER-HDには従来のハードディスクインタフェイスに代わって正式なSCSI (Small Computer System Interface)が採用されている。すでにHuman68kではVer.2.0より大容量メディアへの対応をすませていたが，従来のハードディスクインタフェイスでは，80Mバイトをひとつのドライブとして使うことができなかった (40Mバイト×2台としては使える)。それが，SCSIの採用により，80Mバイトのハードディスクはもちろん光磁気ディスクだろうが，DATだろうが，CD-ROMだろうが，接続できることになったわけだ。

80Mバイトのハードディスクを内蔵したX68000SUPER-HD。カラーはシリーズ最高級機にふさわしいチタンブラックが採用され，マシンへのこだわりを表現してくれる。ディスプレイはCZ-613Dタイプのみチタンブラックカラー(CZ-613D-TN)を用意しているので，ぜひとも組み合わせたいところだ。

もちろん，SUPER-HDにしてもこれまでのX68000と完全にコンパチブルだが，外付けのハードディスクをさらに増設する場合には従来のX68000対応のものではなく，SCSI対応のものを選ぶ必要がある。

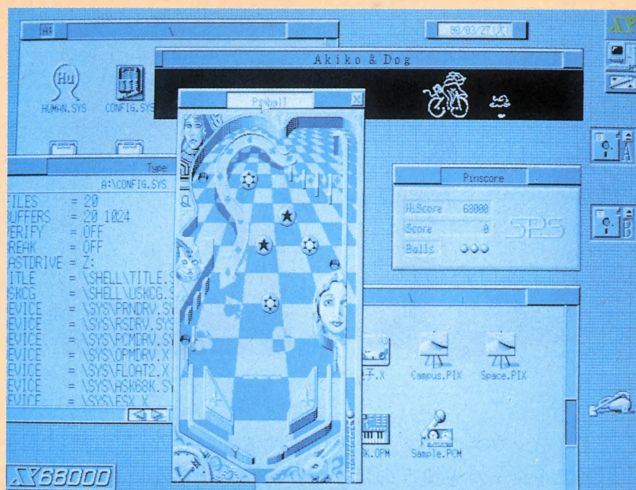
PROⅡ & EXPERTⅡ

EXPERTⅡシリーズとPROⅡシリーズに関しては，SX-WINDOWが搭載され，後ほど触れるBIOSの高速化以外には，これまでのEXPERT/PROと基本的に変わらない。ハード的にはまったく同じだと考えていいだろう。

EXPERTシリーズは初代X68000のデザインを継承する縦置きタイプ，マンハッタンシェイプ，ツインタワー，そしてポップアップハンドルである。もちろんマウスはマウストラックボールである。メインメモリは2Mバイト。

PROシリーズは新たなスタンダードとして期待される横置きタイプ。本体が大きいぶん，拡張スロットが4つあるのが最大の魅力である。キーボードも大きいが使いやすく，重量はむしろEXPERTのものよりも軽い。あとはユーザーの用途や好みによるが，立体視端子がないことと，マウスがトラックボールに変形しないふつうのマウスであるなどの違いはある。

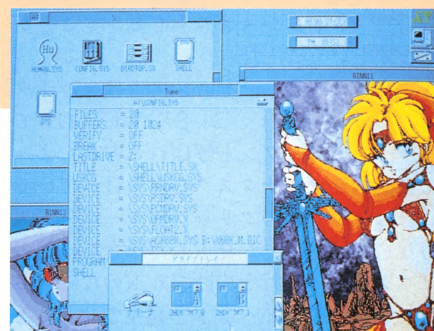
さて，PROⅡシリーズの場合，メインメモリは標準では1Mバイトだが，これは増設すればEXPERTシリーズと同等の機能となるので問題はない。製造工程が楽なためか，あるいは差別化のためかEXPERTシリーズとの価格差が53,000円ある。1Mバイトの増設メモリが純正のものでも28,000円と安くなったぶんPROシリーズのお買い得度はアップしたといえるだろう。

これがSX-WINDOW。4階調表示のウィンドウやアイコンはNeXTもまっさおのカッコよさ。標準で付いてくるサンプルのピンボールがまたよくできている。また，右下の画面では16色(65536色より選択) モードのグラフィックを表示。キャンバス.Xをダブルクリックするとグラフィック用のウィンドウが開くので，そこにグラフィックデータのアイコンを放り込むだけ。ちなみに絵はX68000ユーザーでもある森林林檎氏だ。

なお，EXPERTⅡ/PROⅡの従来機種との見分け方だが，本体正面にあるX68000のロゴマークを見ればよい。これまで表面に印刷されていただけだったのが，なんと豪華な（？）バッジになっているのだ。

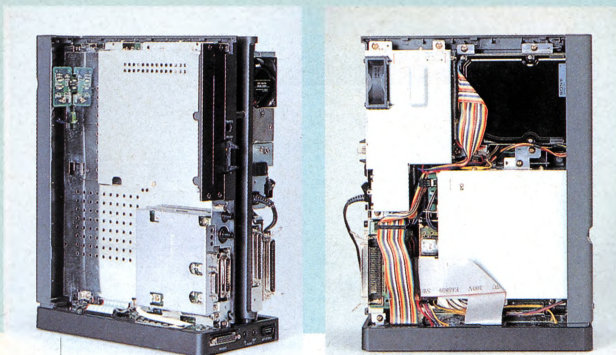
SX-WINDOW

ではいよいよ肝心なSX-WINDOWの解説に移ろう。はっきりいって，今回の新製品ではハードウェアよりもウィンドウシステムを搭載したことのほうが重大である。

SX-WINDOWは，Human68k上のウィンドウシェルでビジュアルインタフェイスとして新たな環境を提供するものだ。一見してビジュアルシェルを強化したものと思われるかもしれないが本質的にはまったく違う。アプリケーションを実行させる環境としてのウィンドウシステムなのである。

簡単にいうと，このウィンドウシステムの上で動くアプリケーションを作れば，複数のアプリケーションを統一的な操作のもとで，呼び出して実行させたり，べつのアプリケーションに移動したりすることができ，ものによってはアプリケーション間でデータのやりとりを行うこともできる。

たとえば，写真のピンボールはSX-WINDOW上で動くサンプルプログラムで，その後ろのウィンドウでは「暁子.X」という女の人が自転車で走る別のプログラムが動いている。「暁子.X」を実行するには，そのプログラムの入ったファイルウィンドウを開いてそこに表示される「暁子.X」を表す

オリジナルイメージをストレートに保持するX68000EXPERTⅡシリーズ。
写真はCZ-613C-BKとCZ-602D-BKの組み合わせだ。

横置きタイプのX68000PROⅡシリーズ。写真はCZ-653C-GYとCZ-603D-GYの組み合わせ。新たなスタンダードとして定着しそうだ。

いちだんとスッキリしたSUPER-HDの内部。右側の黒いユニットが80Mバイトのハードディスク。こんなに小さいのだ。

アイコンをマウスでダブルクリックする。ビジュアルシェルと違うのはここから。つまり単にプログラムを呼び出すのではなく，このウィンドウシステム上で動かすことができるということだ。

ご存じのように，Human68kはマルチタスクのOSではない。が，SX-WINDOWではイベントドリブンと呼ばれる方式によって疑似的なマルチタスク処理を行っている。これは，マウスのクリックやキーボード入力などのイベントが発生した場合には他のプログラムの実行を中断するものだ。プログラムは自分でイベントを発生させることもでき，これによって，走っている暁子さんを見ながら別のウィンドウを開いてピンボールを楽しむこともできるわけだ。

SX-WINDOWではテキスト画面にモノクロ4階調とカラー3色（RGB）を使用し，これに加えて65536色中16色のグラフィック画面を使用できる。グラフィックの場合，パレットの設定はアクティブなウィンドウ（いちばん上のウィンドウ）について有効となる。パレットの異なるグラフィックを使用したウィンドウをアクティブにするとバックにまわったウィンドウのパレットは一時的に壊れる。

ウィンドウの情報はメインメモリに保持され，メモリの余裕さえあれば同時にいくつものウィンドウを開いてアプリケーションを走らせることができる。逆にいえば，それだけメモリを必要とするわけで，残念ながら1Mバイトタイプの機種では増設しないとほとんどなにも動かすことができない。ゲームにしか使わないのであればべつだが，PROIIシリーズの場合には可能な限りメモリを増設しておきたいところだ。

さて，SX-WINDOWは，ウィンドウ環境を実現するSX-SYSTEMと，操作環境を提供するSX-SHELLからなる。SX-SHELLではビジュアルシェルと同じく誰にでも感覚的にファイル操作ができ，さらに，いままではコマンドモードでないとできなかった機能のうちかなりの部分がサポートされている。これらについては，121ページからの「これがSX-WINDOWだ！」で吉田幸一氏が詳しい解説を行っているのでそちらを参照されたい。ここでは，ウィンドウシステムの要となるSX-SYSTEMについて触れておこう。

SX-WINDOWではウィンドウ上に表現されるメニューやアイコン，グラフィックデータなどのさまざまな資源を複数のアプリケーションで有効に利用しようという，リソースの概念が採用されている。どういうことかというと，アプリケーションはプログラムコードとウィンドウ上で扱うデータ（リソース）を分離して持ち，そのリソースの管理をSX-WINDOW側に持たせるというわけだ。

SX-WINDOWは表示のための単なるウィンドウマネージャではなく，SX-WINDOW上のプログラム作成を支援するためのシステムでもある。そしてウィンドウ上のプログラムが必要とする基本ルーチンをシステム内に持っている。ちょうど，Macintoshのツールボックスのようなものだと考えればよい。もちろんそれらのルーチンはオリジナルだが，そのファンクションコールはMacintoshのツールボックスとコール番号や呼び出し方を合わせているようだ。このため，Macintosh用ソフトの移植や，同時開発も容易となるだろう。

SX-WINDOWはディスクによって供給され，デバイスドライバの形で登録しHuman68kの機能を拡張する。このため，新製品のみならずすべてのX68000でSX-WINDOWを利用できるわけである。シャープでは6月ごろにこのSX-WINDOWを別売りし，従来機種をサポートする予定で，価格も1万円以下に抑えたいとのことだ。

また，今回の新製品からBIOSの一部が高速化（平均2倍）されているが，これはROMが変わったわけではなくIOCS.Xを登録することによって拡張部分をRAM上に持つ。主に高速化されたのはグラフィック関係のBIOSが中心のようだ。

周辺機器も充実

X68000の新機種については以上でおくとして，本体以外にもいくつかの周辺機器が発表されているので順に見ていこう。

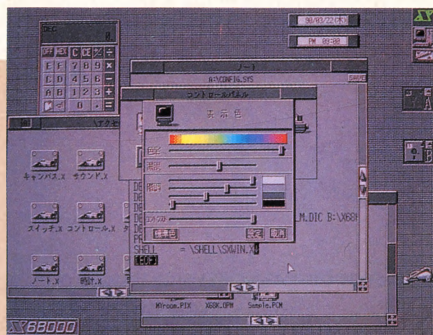
●15型ディスプレイテレビ

まず，ディスプレイから。今回は2機種で，ドットピッチ0.39㎜の高精細度タイプCZ-605Dが115,000円，ドットピッチ0.31㎜の超高精細度タイプCZ-613Dが135,000円だ。いずれも音声多重デコーダ内蔵の15型ディスプレイテレビで，3W×2のステレオアンプを内蔵，着脱可能な外部スピーカも装備している。当然，水平周波数31/24/15kHzの3モードマルチスキャン。リモコンとチルトスタンドも付いている。

色は基本的にオフィスグレーとブラックの2色だが，CZ-613DだけはX68000SUPER-HDに合わせてチタンブラックを選ぶ

光磁気ディスクユニットCZ-6MO1。595Mバイトの大容量を誇る外部記憶装置だ。

●新製品一覧

名　称	型　番	価　格	発売日
X68000SUPER-HD	CZ-623C	498,000円	6月1日
X68000EXPERT II	CZ-603C	338,000円	3月15日
X68000EXPERT II-HD	CZ-613C	448,000円	3月15日
X68000PRO II	CZ-653C	285,000円	4月15日
X68000PRO II-HD	CZ-663C	395,000円	4月15日
15型ディスプレイテレビ	CZ-605D	115,000円	3月15日
	CZ-613D	135,000円	3月15日
光磁気ディスクユニット	CZ-6MO1	価格未定	6月
SCSIボード	CZ-6BS1	価格未定	6月
ビデオボード	CZ-6BV1	価格未定	6月
1Mバイト増設RAMボード	CZ-6BE1B	価格未定	4月
SX-WINDOW	CZ-259SS	価格未定	未定

ビジュアルシェル同様にウィンドウの表示色も自由に変えられる。X68000であるからには当然だね。

ことができる。

●光磁気ディスクユニット

ついに光磁気ディスクユニットがX68000用にも登場した。光磁気ディスクは書き換え可能な大容量メディアとしてもっとも期待されているもので，今回発表されたCZ-6MO1は完全なシャープ純正品。直径5.25インチのディスクを採用し，記憶容量は595Mバイト。記録フォーマット，ディスクカートリッジともISO規格を採用している。データアクセスも高速で，平均シークタイムは60㎳秒，データ転送速度は925Kバイト/秒となっている。

この光磁気ディスクをX68000で利用するにはインタフェイスとしてSCSIが必要となる。そこで，オプションのSCSIボードCZ-6BS1も同時に発売されることになった。もちろんSUPER-HDの場合にはSCSIが内蔵なのでそのまま接続することが可能だ。

●ビデオ出力ボード

X68000の画像をビデオに録画するには，RGB信号をNTSC信号に変換しなければならない。ビデオボードCZ-6BV1はそのためのもので，ゲーム画面やCGアニメーションなどを手軽にビデオに録画することができる。これまではカラーイメージユニットを使わなくてはならなかったが，デジタイズ機能やスーパーインポーズ機能が必要ない人には，このビデオボードで十分だろう（なお，このビデオボードにはS端子も付いている）。

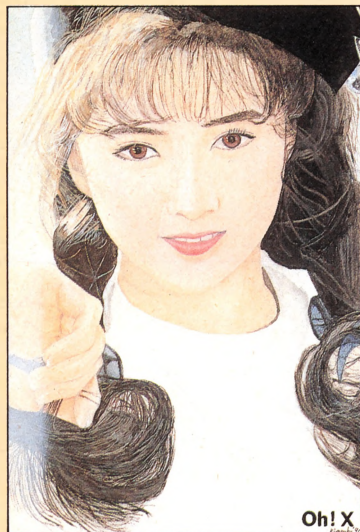
＊

X68000もいよいよ4年目の春を迎えた。ユーザーにとっては基本仕様の変更が気になる時期でもある。クロックは16MHzに？32ビットはどうなる？　と，噂はいつも先走りする。が，賢明な読者の予想どおり今回もハード的な基本仕様の変更は一切ない。X68000は16ビットパソコンとしての地固めを着実に行っている。その成果のひとつが今回のウィンドウシステムであったりするわけだ。5年間は仕様を変えないといったことがいよいよ大きな意味を持ってきたといえるのではないだろうか。　（編集部）

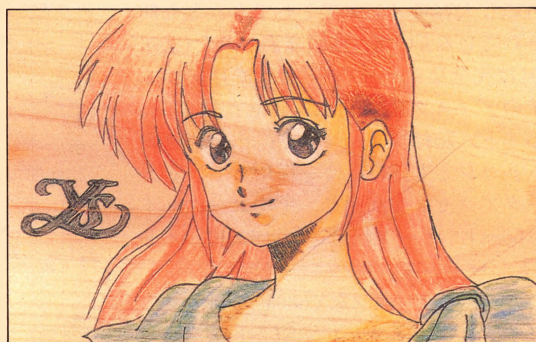
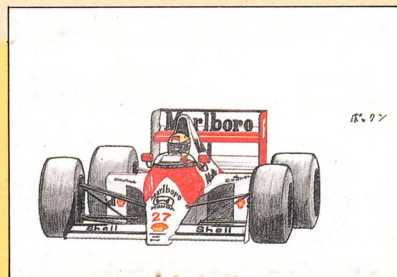
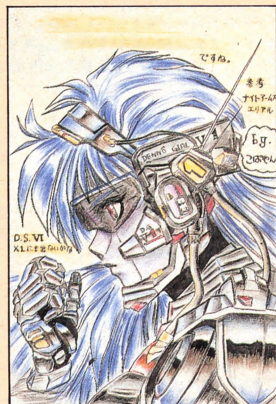
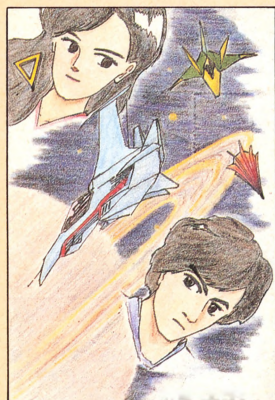
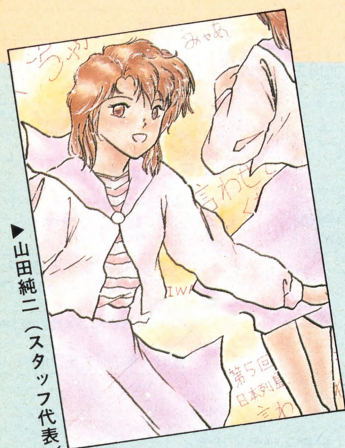
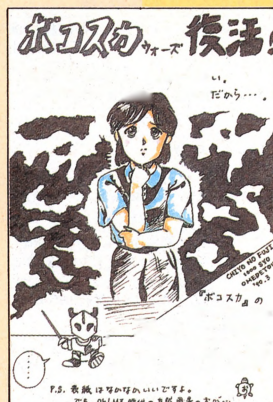
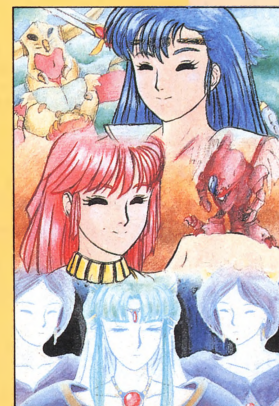
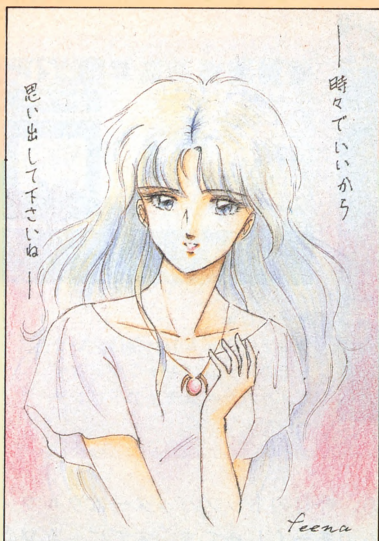
●X68000SUPER-HD/EXPERTⅡシリーズ/PROⅡシリーズ仕様

	SUPER-HD	EXPERTⅡシリーズ	PROⅡシリーズ
CPU	68000（10MHz） 80C51（キーボードスキャン/テレビコントロール用）		
ROM	IPL，BIOS　128Kバイト キャラクタジェネレータ　768Kバイト 16×16ドット，24×24ドット　全角（JIS第1/第2水準漢字） 8×16ドット，12×24ドット　半角 8×8ドット，12×12ドット　1/4角		
RAM	メインメモリ　2Mバイト （最大12Mバイト）		メインメモリ　1Mバイト （最大12Mバイト）
	テキスト用VRAM　512Kバイト（ビットマップ方式） グラフィック用VRAM　512Kバイト（ビットマップ方式） スプライト用VRAM　32Kバイト スタティックRAM　16Kバイト		
表示能力 実画面エリアサイズ	テキスト　1024×1024ドット　4プレーン グラフィック　1024×1024ドット　4プレーン 512×512ドット　16プレーン		
表示能力 表示画面モード テキスト表示	●実画面エリア　1024×1024ドット時 高解像度モード　768×512ドット 512×512ドット 512×256ドット 256×256ドット 標準解像度モード（オーバースキャン）　512×256ドット 256×256ドット 512×512ドット（インターレース） 各モードともドット毎に65536色中任意の16色指定可能		
表示能力 表示画面モード グラフィック表示	●実画面エリア　1024×1024ドット時 高解像度モード　768×512ドット 512×512ドット 512×256ドット 256×256ドット 標準解像度モード（オーバースキャン）　512×256ドット 256×256ドット 512×512ドット（インターレース） 各モードともドット毎に65536色中任意の16色指定可能		
	●実画面エリア　512×512ドット時 高解像度モード　512×512ドット 512×256ドット 256×256ドット 標準解像度モード（オーバースキャン）　512×256ドット 256×256ドット 512×512ドット（インターレース） 各モードとも(1)ドット毎65536色中任意の色指定可能（1面） (2)ドット毎65536色中任意の256色指定可能（2面） (3)ドット毎65536色中任意の16色指定可能（4面）		
表示能力 表示画面モード スプライト	●パターン定義 サイズ：16×16ドット/パターン，8×8ドット/パターン 定義数：128パターン（バックグラウンド2面未使用時最大256パターン） 色　：1パターンにつき16色/65536色（ドット単位） ●表示 スプライト座標系：1024×1024ドット 表示画面：512×512ドット（バックグラウンド1面表示） 256×256ドット（バックグラウンド2面表示） 表示制限：128スプライト/画面，32スプライト/ライン		
表示能力 特殊機能	スムーススクロール（テキストは円筒。グラフィックは球面）/特殊画面制御機能/プライオリティ機能/パレット機能/半透明機能/スーパーインポーズ機能		
サウンド機能	FM音源：ステレオ8オクターブ　8重和音同時出力 音成合成：AD PCM（Adaptive Differential PCM）		
ハードディスクドライブ	CZ-623C：80MB3.5″HD内蔵	CZ-613C：40MB3.5″HD内蔵 CZ-603C：　〃　内蔵可能	CZ-663C：40MB3.5″HD内蔵 CZ-653C：　〃　内蔵可能
フロッピーディスクドライブ	1Mバイトタイプの5″2HDフロッピーディスクドライブ2基搭載 （オートロード/オートイジェクト）		
入力装置	ASCII準拠フルキーボード		
	マウストラックボール同梱		マウス同梱
インタフェイス	プリンタ（セントロニクス社仕様に準拠）/ジョイスティック（2個）/テレビコントロール/アナログRGB出力/オーディオ出力/RS-232C/外部フロッピーディスク/マウス/イメージ入力端子		
	立体視端子		―
	SCSIインタフェイス	ハードディスクインタフェイス	
拡張I/Oスロット	2スロット内蔵		4スロット内蔵
電源・消費電力	AC 100V 50/60Hz CZ-623C：47W（待機時6W以下）	AC 100V 50/60HZ CZ-613C：47W（待機時6W以下） CZ-603C：38W	AC 100V 50/60Hz CZ-663C：42W（待機時5W以下） CZ-653C：35W
外形寸法・重量	本体：幅155×高さ363×奥行270㎜ CZ-623/613：9㎏，CZ-603：8㎏ キーボード： 幅463×高さ33×奥行196㎜ マウストラックボール： 幅73×高さ32×奥行105㎜		本体：幅430×高さ128×奥行340㎜ CZ-663：12.7㎏，CZ-653：12㎏ キーボード： 幅480×高さ40.9×奥行221㎜ マウス： 幅63×高さ37×奥行97㎜
付属ソフト	オリジナルウィンドウシステム（SX-WINDOW ver.1.0） オリジナルOS（Human68k ver.2.0） オリジナルBASIC（X-BASIC ver.2.0） 辞書ディスク ver.2.0，日本語ワードプロセッサ，辞書ユーティリティ		

カラーイラスト大集合
Oh!X readers'ぎゃらりぃ

さあ，今年もやってきました「言わせてくれなくちゃだワ」。というわけで本文に先がけ皆さんからのカラーイラストをどーんとご紹介しましょう。それにしても数年前には考えられなかったパワーですね。そうそう，135ページのイラスト大賞もよろしくね。

▲鳥羽恭暢（愛知県）

▲小井田伸雄（岩手県）

実

▲高木智之（神奈川県）

▶山田純二（スタッフ代表）

宏（兵庫県）

▲伊藤浩克（香川県）

▲菅原真希子（秋田県）

▲見浦　崇（長野県）

▲井上敬介（神奈川県）
受験勉強で使った鉛筆を貼り付けて作ったX1turboのロゴマーク。正に勝利のシンボルですね。井上君，合格おめでとう！

▲味野真一（岡山県）

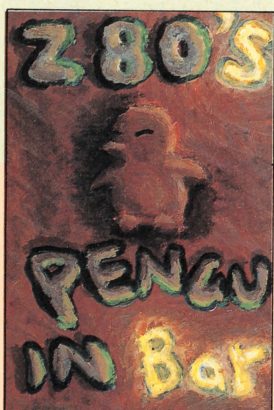
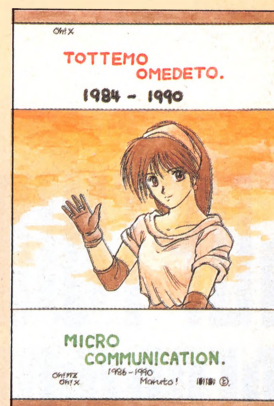
▲丸藤俊之（神奈川県）

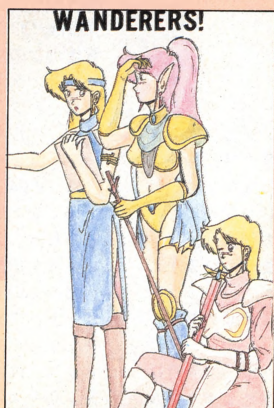
▲森下　保（静岡県）

▲広瀬晃司（滋賀県）

▲迎谷彰信（茨城県）

▲玉野健一（奈良県）

ぱーそなる こんぴゅーた まがじん
おー!! えっくす

▲清水健年（東京都）

大久保益幸（滋賀県）

▲高橋弘幸（神奈川県）

▲浅田善之（大阪府）

▲渡辺光輝（埼玉県）

THE SOFTOUCH

SOFTWARE INFORMATION

あーくしゅ
ウルフ・チームの新作は，いままでのゲームのキャラがデフォルメされたパロディアドベンチャーだ。

SOFTWARE information

やれやれ，やっとあたたかくなってきました。皆さん，元気にゲームしてますか？　今月はちょっと新しいゲームが少なめですが，力の入ったものが揃っています。こりゃ，夏以降も期待が持てそうだね。

話題のソフトウェア

桜もさっさと咲ききっちゃって，なんとなく4月らしくない日が続いていますが，みなさまいかがお過ごしでしょーか。

さて，この春休みはワンダラーズ・フロム・イース，ポピュラス，ダンジョンマスターと大物が目白押しだっただけに，暇を持てあましたゲーマーは少なかったんじゃないかナ。うんうん，いいことだ。これからもどんどんこういったゲームが出てほしいもんだ。

てなことで，今月も新しいゲームを紹介していくわけだけど，今月はいつもよりやや数が少なめです。まあ，ソフトメーカーさんも夏に向けて中休みってとこなんでしょう。きっとひそかにゲームを開発していると信じていますわ，ホホホ。

さあ，今月のトップバッターはウルフ・チームの**あーくしゅ**です。すでにもう発売されているからプレイした人もいるかな。このゲーム，いままでのウルフ・チームのキャラクターたちが総動員されているパロディアドベンチャーなのです。キャラクターたちは，それぞれかわいくデフォルメされていて，いままでのウルフとは違った魅力があります。そして，なんといっても“じぇだ”のあっぱらぱぁな言動は感動モノ。思わず画面に向かって“タぁ～コ”と叫んでしまうほど。ほのぼのとしたい人にはオススメの1作です。

お次はシステムサコム。ここからは2作品が出る予定です。まずひとつは**ジェミニウイング**。このゲームは，ファミコンやゲーセンでお馴染みのテクモのアーケードゲームからの移植で，昆虫をモチーフとした敵キャラをガンガン打ちまくるシューティングゲームなのです。来月には詳しいこ

“地下迷宮主人”真強的！

順位	タイトル	前回
1	ダンジョンマスター	1
2	ワンダラーズ・フロム・イース	7
3	バブルボブル	—
4	サンダーブレード	—
5	ソーサリアン	3
6	ポピュラス	—
7	A-JAX	—
8	三国志Ⅱ	—
9	スーパーハングオン	4
	ジェノサイド	8

おめでとーございます！　ダンジョンマスターが2位の3・倍・近・い・得票（!!）を得て首位を守りました。私が担当したなかでは，最高の突出ぶりです。前評判だけじゃないことを見事に証明してくれました。この独走は，ワンダラーズ・フロム・イースでも止められるかどうか。同時期発売のポピュラスが，けっこう喰らいついてくるかもしれない。日米欧の対決となるとまた面白いんですが。

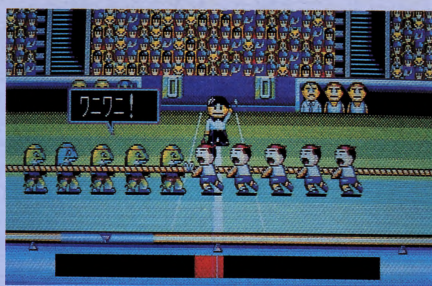
このところ生産力がすごいのがSPS。今月は3作をランクインさせました。順位こそバブルボブルに押さえこまれたものの，最近「いい移植をする」とブランド力も上昇中。電波との高次元での勝負が期待されます。

さぁて，あのロングセラー「三国志」に後継者が登場。X1ユーザーの力で8位に入ってきました。これは長く居座りそうだぞ，新たな「ソーサリアン」となるか？

それにしてもなんて豪華な顔ぶれなんだろう。圏外だってこんなにすごいのに。というわけで，初公開，11位～16位（こっから下はどんぐりの背比べなのだ）のランキング，いってみよう。

順位	タイトル
11	メタルサイト
12	アフターバーナー
13	アルガーナ
	ファーストクイーン
15	V'BALL
16	テトリス

おおっ，こんなところにメタルサイトが！　先月2位だったのに……。アフターバーナーも粘っているぞ。それにSPSがまた1本。アルガーナとファーストクイーンはいいところまで来ながら，浮いたり沈んだりだ。そして定番テトリス。11位以下も充実してるなぁ。こりゃあXシリーズの未来も明るいな。はっはっは，ということで，また。　（浦）

アソコの幸福

タッグ・オブ・ウォー(画面はPC-9801版)

サーク

とを紹介できそうなので，もうちょっと待っててね。で，もうひとつはサコムお得意のノベルウェアもの。タイトルは**闇の血族**。これは，サコムがこれからシリーズ化していく“名探偵魅由”の第１弾で，主人公となる美少女“魅由”が事件を解決すべく活躍するミステリーアドベンチャー。こちらも，もうちょっとしたら詳しいことが載せられそうです。

A-JAXで一躍脚光を浴びたコナミからは，シューティングとパズルの要素をあわせ持つ**クォース**が，もうすぐ発売される予定。これは，上から落ちてくるブロックにバシバシ弾を打ち込んで，四角くして消していくというパズルゲーム。なかなか奥が深いので，いろいろなテクニックを磨いて楽しんでほしいな。

さて，ねじ式が好評だったツァイトでは，**アソコの幸福**の開発が着々と進行中。とりあえず開発中の画面をお届けします。詳細はもうちょっと待った！

ザイン・ソフトでは綱引きゲームとでもいえばいいんでしょうか，**タッグ・オブ・ウォー**です。開発も順調のようでもうすぐ発売の予定です。お楽しみに。

そして，めぞん一刻などでお馴染みのマイクロキャビンからは，RPG**サーク**が登場です。ひさびさの大作だけに，力が入っているようですね。出来上がりが楽しみな１作です。

で，先月このページで紹介したM. N. M Softwareの**LIFRAIM**（先月LIFRAINと書いたのは間違いです。ごめんなさいっ）ですが，ようやっと画面をお届けできるようになりました。このゲーム，チーズをうまくドアまで運んでいくわけですが，途中で失敗してチーズを落としたときのネズミの表情がたまらなくかわいいっ！　全部で50面以上用意されているようです。

そして，新規参入会社アミューズメントからはアクションRPG**ブレード・オブ・ザ・グレート・エレメンツ**が発売される予定です。とりあえずは画面写真だけですが，期待して待っていてください。

さて，ここで発売中のゲームを紹介しなくてはね。X1ユーザーにはもう心の友となったソーサリアンシリーズの最新作**ギルガメッシュソーサリアン**，そして**三国志Ⅱ**が発売されています。三国志Ⅱに関しては，また来月ドドッと紹介するつもりですのでお楽しみに。そうそう，**Misty4**もすでに発売中です。シブくゲームにひたりたい人にはうってつけですね。ま，今月はこんなところかな。数が少なくてゴメンナサイ。来月はきっともっと載せられると思うのだけど……。ではまた，来月会いましょう。

LIFRAIM

ブレード・オブ・ザ・グレート・エレメンツ

いきなりだけどCD紹介

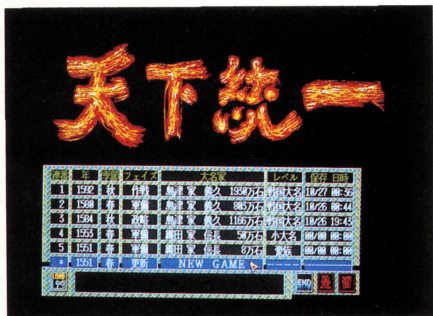
最近アキバあたりのレコード屋に行くと，ものすごい数のゲームミュージックのCDが並んでいる。買うものを決めていっても，そのあまりの膨大さに結局「どれにしようかしら」と悩んじゃう始末（ええい，意志薄弱とでもなんとでもおいい！　ちなみに私は「これがいいわ」と決めるまでまるまる30分かかったことがある）。こんなにCDがあるにもかかわらず，紹介ってしたことないよなぁ，なぁんてふと思って（よーするに思いつき！）作っちゃったのがこのコーナー。まあ，今回限りと思って勘弁してね。

で，何を紹介しようかなって思ってたときに，タイミングよくポニーキャニオンが持ってきてくれたのがこの２つ。アームドF/クレイジークライマー２（20曲/1,500円）と，GAME BOY MUSIC（13曲/1,500円）だ。まずはアームドF/クレイジークライマー２から紹介ね。

ニチブツのアーケードゲーム２作品の曲をすべて収録。ゲーセンでクリアできずに聴けなかった曲もこれで聴ける！　と思って聴き始めたのだけど，最後まで聴いているうちにどーでもよくなってしまった。BGMなのだ，完全に。小気味よいリズム，カンにさわらない音，ゲームをしないで聴くとこんなにも落ち着いて聴けるものか，と思ったほど。気に入ったぞ，私は。

でも，もっと気に入っちゃったのがこっち，GAME BOY MUSIC。いいわぁ，これってば。スーパーマリオランドやらテトリスやら，ゲームボーイ用ソフト４タイトルの曲が収録されているワケなんだけど，いやぁアレンジがお上手，さすがってカンジ。なにしろゲーム臭さがなく，聴いててホントにキモチいいの。

ストリングスを目一杯使ってメロディアスに仕上げている曲もあれば，サックスをふんだんに取り入れたリズミカルな曲もあったりと，バリエーションに富んで飽きさせない構成。そのうえ曲の中でゲームボーイの効果音をこれまたうまぁく取り入れていて楽しませてくれる。ゲームを知っててもゲームミュージックと感じさせない出来，これはもう買いです！　（出口香）

▶受かったぞー，北海道だー！　ところで東京の人は，８畳６畳４畳半風呂キッチンつき月35,000円の家賃をどう思います？　岡村　克宜（18）X68000PRO，MZ-2500　北海道

●天下統一

正統派におすすめのシミュレーション

Kameda Masahiko
亀田 雅彦

PC-9801などで親しまれていた，あの天下統一がいよいよX68000にも登場！ 余計なものを排除し，ゲーム性を重視して作られているこのゲーム，シミュレーションファンでなくても，ぜひプレイしてほしいゲームだ。

X68000用 5"2HD版2枚組 9,800円(税別)
システムソフト ☎092(752)3902

おおーっ！ 天下統一じゃあないか！ 懐かしいなあ。この前プレイしたのは，確か去年の秋だったんだよなあ。あの頃は我を忘れて，夜も昼も忘れて，テスト!? まで忘れそうになりながら，下天を夢見たもんだ。謙信と信玄が死闘を繰り広げ，北条は関東をうかがい，信長が天下をねらう大戦国絵巻が，いま始まる!! あー，なんて感動するストーリー展開。しかも，私がやりたいなあと思ったところへ，ちょうど移植されるこのタイミングのよさ。もはやこれは，私がレビューをやる運命にあったといえよう。いやがうえにも気合いがこもってしまう。今回は，軽快なフルマウスオペレーション，比較的カッチリとしたシステムが，いったいどのようにX68000に移植されたのか？ このゲームはたいへん面白い！ という（独断的）絶対評価といっしょに，ドドーンと解説しちゃいましょ。

第一絵巻 まだまだ豪族レベル

はじめに断っておきますが，私はこのゲームがとっても気に入っているので，それを心して読むように。

まず，題名からもわかるように，これは戦国ものだ（もう知ってる？）。発売元は，あのSUPER大戦略などでお馴染みのシステムソフト。シミュレーションゲーム界の大御所。といっても，今回移植を担当したのは，アルシスソフト。システムソフトとX68000とはあまり馴染みがないから，まあ妥当な線かなあ（決して，よかったなどといってはいけません）。でも，きれいに仕上がってるようで，よかった。よかった。

天下統一では，軍備・政略・作戦・合戦フェイズが，1年に4回フォーシーズンまわってくる。兵や鉄砲を集め，よくある内政（開墾など）・外交（同盟など）コマンドを各フェイズで実行。合戦時には別画面へと移行して，敵部隊との戦闘か城攻めとなるパターン。また各季節の前には，大雪・山師（鉱山発掘だ）・疫病（コロリか？）なんかが起こる地方もある。でも，こんなのは，その辺にころがってるシミュレーションと同じで重要じゃない部分。そこで，突如として箇条（過剰）書きレビューへと突入するのだった。

題して，**ここが違う天下統一！**（または，ここが好きやねん！ 天下統一）

その1：なんといってもマップが特徴的。昔からの，「でっかい国(地方)の奪い合い」じゃなくて，国の中にいくつかの城があって，それをひとつずつ勝ち取って，その国を支配し，全国を統一する方式。つまり，「越前国の一乗谷と北ノ庄だけを支配している」なんてことがあるわけ。もちろん内政で，超でっかい城をつくることも可能。

その2：信玄公曰く，「人は城，人は石垣，人は堀」というように，登場する武将も桁違いに多い（なんでも825人らしい）。それに，武将が死ねば，そのあとをついで新たに武将が登場するので（このときの名前はランダム），無数にいるといってよい。設定値は，年齢・軍事・内政・忠誠・兵士数・鉄砲数の6個だから，ちょうどいいくらい。そこで，天下統一ならでは！ というのを探すと，このAV時代にありながら，武将の顔のグラフィックというものがないのだ。そこで，私は気づいてしまった。いままでの戦国ものに，必ず違和感を抱きながらプレイしていたこと。そして，その原因がこのグラフィックであり，それらは必ず見飽きるということに……。

その3：戦闘シーンの簡素化。両軍3部隊ずつ出て（別動隊もある），鶴翼の陣！ 魚鱗の陣！ とかいいながら(Z△Nにもあったなあ)，長篠の戦いしたり（鉄砲だよん），白兵戦したりする。このシステムには，やれ地味だ！ やれ簡単だ！ と賛否両論がうずまいていた。が，このゲーム全体のコンセプトからして，これはこれでバッチリだと思う。つまり，私は好きだ！ の鶴のひと声でおしまい。

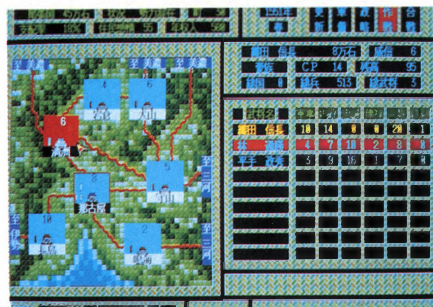
ああ，なんか肩に力が入ってしまった。ようするに，城と武将の2つは大事なのよん！ ということ。

第二絵巻 ここまでくれば小大名

さて，基本システムはわかってもらえた。そんなこんなで，実際のゲームはどうなっているのか？

最初に，マウスオペレーションだと書いた。X68000では当たり前だが，なかなか好調のもよう。YES・NOが「是・否」なのはご愛敬か？ 日本地図を直接指して，ここ！ なんていう操作ができるのは快適ライフなのだ。

コマンド決定画面

▶東欧もソビエト連邦も変わりましたね。中国は相変わらずだけど。
澤 英敏（18）X68000EXPERT 北海道

そしてゲーム開始時には，全国の大名に，その石高に応じてあるレベルが設定されている（もちろん，ゲーム中にそのレベルはアップする)。それが，中見出しにもなっている「豪族・小大名・戦国大名」の3ランク。1国も支配してないと豪族，1国で小大名，100万石で戦国大名になる。このレベルアップがはじめの頃の目標で，これまたシンプルなわりに感動するんだ。

それともう2つ。コマンドポイント（以下CP）と「威信」ていうパラメータもある。CPはようするに，自分が行動できる量だ。信長なんか超行動的だから，この値も大きい。

威信は，わかったようなわかんないような言葉だけど，とりあえず「外交や勝利条件に関わるよ」ぐらいでいいのだ。威信が高ければ，58カ国すべて支配していなくても，天下に号令できちゃうのだ（逆に同盟破棄で威信が下がる)。この威信を上昇させるのが，なんと朝廷からの官位授与システム！　大名がへへーっと頭を下げて，かしこくもミカドより官位を賜わる。「上杉謙信を越後守護に任命する」などという，ありがたい詔勅が下されることであろう。なお，朝廷ばかりでなく，異国の南蛮船が来たりもする（鉄砲持ってるのだ。南蛮人恐るべし)。

このように，朝廷が登場するのは，あの筋の方ならずともうれしい限りである（ちなみに，編集部と靖国神社&皇居は隣接しているが無関係？)。戦国時代の朝廷は権力こそなかったものの，その超越的権威は無視しきれなかったはず。秀吉が，将軍職より関白を欲しがったのも，このためだそうだ。これでやっと枕を高くして眠れるというもんだな。なお，「鎮守府将軍」という官位があるが，いわゆる将軍のことじゃない。知ってるよね。

朝廷より，バシバシと官位を賜わるようになる頃には，そろそろ戦国大名に手が届くんじゃないかな。そんなときに心してやるべきことは，実は領内経営だったりする。やみくもに戦うだけでは，決してすぐれた武将にはなれないのだだだ!!　治水開墾・楽市楽座・城の普請（建て増しのこと）で，民に愛される「お館さま」になれ！（ちなみに，楽市楽座とは，信長独自の政策ではない）

で，領国経営が大事なのは，兵の募集があるから。有能な武将を集めるのもさることながら，兵を目一杯集める（50が最高）くらいでないと，戦国大名にゃなれないわな！

ビジュアルもなかなか美しい

おのおのの武将のパラメータ画面

第三絵巻　戦国大名への道

こんなところを踏まえておけば，もはや我が覇道に妨げなし！　当たるものすべてを破壊してくれる。なんて，うまくいけばいいけどね。実際にはほんと難しい（武将によっては簡単だけど)。だって，武蔵国の城なんてみんなレベル10以上だし，なんとレベル40！　なんて，もう勘弁してよといいたくなるような城もある。こういうの，全国の城の中から見つけ出すのって楽しいね。

さて，領国もどんどん増えて，見ていると目がうるうるしてくるようになれば，天下はもうすぐそこだ！　そこで，天下統一目前のプレイヤーは，必ず「最後の決戦」を強いられることになるだろう。その相手はどこかわからないが，複数の強大な大名の戦いになるのが天下統一の慣例なのである。これも，基本的なゲームシステムがすばらしいおかげ。まるで関ヶ原のような気分なのだ。はたして，勝利の女神はいずこに微笑むのだろうか!?

やっぱ天下統一は面白いや。派手なグラフィックとか，ものすごいアニメ処理とかないけど，統一された思想がビンビン伝わってくる。それはいったい何かなあ？　と考えてみると，ボードゲームに行きつくんじゃないかな。派手さをウリにできないボードゲーム界じゃ，プレイヤーが「考える」ということに重点がおかれる。それに対してコンピュータゲームは，どうしても見た目の勝負だ。天下統一は，ほかの戦国シミュレーションといわれるジャンルとはまったく別のジャンルなのだ。変なところで感心してしまった。

第四絵巻　そして，天下統一

このゲームは，露骨に好き嫌いが表れる。コンピュータシミュレーションに対して，何を望んでいるか（グラフィックか思考ゲームか）で評価がまっぷたつに割れるのだ。それに，実は遅い（おおーっ！　恐れていたことが……）というのもあるような気がする。真面目な思考ルーチンだと，もろCPUの速さが出ちゃうんだよう。でも，PC-9801から進化したところもあって，降伏した武将は捕らえなくてもよくなっている。考えようによっては，毎日少しずつプレイしていって，息長く遊べるようになったともいえる。私みたいに飽きっぽい人間には，かえって戦国時代の「もののふの心」がわかるというものだろうか？

浅井長政の居城小谷城とか，高天神・掛川・二俣城とか，二条城・比叡山と聞いて，なんで有名だったかすぐわかる，あるいはNHKの大河ドラマは欠かさず見ている！という剛の者，大戦略のディスクはもうすりきれてしまったという人なら，天下統一は絶対に面白いはずだ。私が保証する！

ある天下統一のプレイ状況

1：「信長が強いからひとりで清洲を落として，平手のじいと林で鳴海を落とそう」

「まあ順当にきたな。でも伊勢長島の独立勢力が意外にしぶとくて，年内の国内平定は無理みたい。無念。やっぱり雨で鉄砲が使えなかったのが痛かったなあ」

2：「あーっ！　今年の冬も大雪だ！　そんなバカな。全然動けないじゃん。これじゃ謙信・柿崎の武力も宝の持ち腐れだね」

このように，配下の武将を最も効率よく配置につけるよう工夫するのが，天下統一の醍醐味。つまり，大戦略ライクなわけですよ。マップとしては城コンセプトの導入，戦闘シーンの簡略化などにみるべきものがある。

戦国時代に思い入れの強い人間は，かえってシンプルなゲームのほうが楽しめる。もちろん，ゲームとしてのシステムがしっかりしている大前提は当たり前だ。天下統一は，ゲームとしての完成度が非常に高いソフトといえる。

(本格)シミュレーション	9
戦国もの	6
城のコンセプト	8
武将の多さ	7
移植もの	7
ハマり度	10

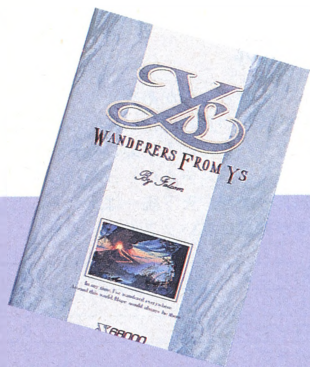
▶ゲームのみをやっているX68000ユーザーに告ぐ，君のX68000は泣いているぞ！　ゲームだけをするなら，私のメガドライブと交換しよう！

羽生　知浩（17）MZ-1500　北海道

●ワンダラーズ・フロム・イース

マドルのイースIII 冒険記〈中編〉

Nishikawa Zenji
西川 善司

先月に引き続き，アドルとはまったく関係ないマドル＝リステリンがお届けするイースIIIの冒険記。アクション性がアップしたため，なかなか先へ進めない人も，これを読めば大丈夫……かなぁ。

X68000用　5″2HD版4枚組　8,700円(税別)
日本ファルコム　☎0425(27)6501

燃える翼竜・ギャルバくん

エレナ「無事だったのね。よかった」

灼熱のマグマ地帯からやっとのことで脱出した俺は，イルバーンズの遺跡の中をさまよい歩いているエレナとはち合わせた。

マドル「なぜ君はこんな危険なところに」

俺はマドル=リステリン。あの有名なイースシリーズの主人公アドル=クリスティンのいとこの親戚の友達の近所に住んでいる（まだいってる)。

エレナ「……だ，誰か来たみたい。隠れましょう」

チェスター「いま，このあたりで物音がしたのだが……」

マクガイア王「チェスターどうしたんだ」

チェスター「いま，この辺で物音がしたような気がしたのですが……。私のそら耳だったようです」

チェスターはエレナの兄だが，最近はこのマクガイア王と何やら怪しげなことをたくらんでいるらしい。

マクガイア王「ところでチェスター，ピエールとやらはどうした？」

チェスター「ピエール神父は我々のたくらみに気づいているようです。奴はバレスタイン城の地下牢へ放り込んでおきます」

話声が遠くなっていく……。

エレナ「行ったみたいね……」

マドル「エレナさん，こんな暗闇に俺を連れ込んで。もしかして俺のことが好き……」

エレナ「あなたは状況をちゃんと把握していないのですか!?」

マドル「冗談ですよ。ところでこの部屋は？」

エレナ「あら，あそこに見えるのはなにかしら」

マドル「あ，エレナさん，ひとりで行っては危険ですよ」

エレナ「この壁なんか変じゃない？　石の扉に見えないかしら？」

ぎぎーっ！

マドル「あ，開いた」

エレナ「たぶん，これ影像の在処へ続く道よ」

マドル「(よくわかるなー『キューブランナー(©KONAMI/SPS)』全面クリアの俺にも予想できなかったぜ)。……エレナ，ここから先は危険だ。俺ひとりで行くから，君は町へ帰るんだ」

エレナ「どうして？　あなたをこの遺跡に来させたのは私なんだもの。あなただけを危険な目にあわせるわけにはいかないわ（かなり嫌そうな声で)」

なかなかきれいな容姿のギャルバ君

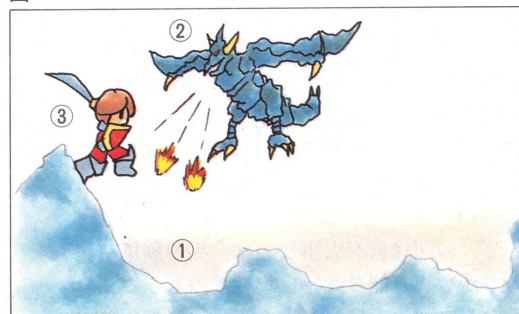
マドル「よし，じゃ，一緒に行こう」

エレナ「きゃーっ，ひとさらいーっ」

一部会話の内容が実際と異なることをお詫びいたします。

ちっ，結局エレナは町に帰ってしまった。まったく最近の娘はわからんぜー（おいおい)。さて，石の扉を開けて少し進むといきなり広い部屋にでる。ここは炎の翼竜ギャルバ君の住みかだったのだ。さあ，装備を確認しよう。武器はロングソード，防具はチェインメイル，スモールシールド，もちろんこれ以上のものを持っていればそれにこしたことはない。そうそう，さっき，エレナが町へ帰ったことからもわかるように，もう町へ戻ることもできるんだよ。装備を揃える前にマグマ地帯へ落とされたアドル君達，ご苦労様でした。いったん町へ戻って装備を揃えてこようぜ（お金もマグマ地帯でたんまり稼いだでしょ)。また，非アクションゲーマーのアドル君達は，さっき挙げた装備より1ランク上のブロードソード(8000GOLD)などを持っていると，いくぶんか楽になるぞ。レベルは7以上あるかな。6でも倒せないことはないが，かなり苦しい戦いになるだろう。さあパワーリングと薬草を装備してギャルバ君と御対面だ。

ギャルバ君は火を自キャラに向かって吹いてくるという単純な攻撃法なのだが，さすがは翼竜，剣の届かないところをふわふ

図1

1．ここで待っていると
2．ここに来るので
3．ここからジャンプして斬る

▶最近の西川善司さんの記事を読んでいると，以前の西川さんと変わってきていることに気づく。下ネタにプラスしてノリがいい。　西本　英樹（17）X68000，X1　北海道

わ飛んでいるため，なかなかダメージを与えることができない。そこで，このマドル様が考えたのが図1の方法だ。まず，いちばんとんがった岩の右側の麓でギャルバ君が来るのを待つ。頭上で止まったのを確認したらすぐさまその岩に上り，そのてっぺんからジャンプしてギャルバ君にデヤーっ！　と斬りかかる。飛降りながら斬る際，着地後にすぐまた岩に登れればもう1回斬ることもできるぞ。

ティグレー採石場，再び

ドギ「あ，マドル。俺はちょっとでかけてくるからな」

マドル「どこへ行くんだい？」

ドギ「昔世話になった師匠に会いにエルダーム山脈へ行く。帰りが遅くなるかもしれないが心配しなくていいからな。そういや，エドガーさんがお前と話がしたいそうだぜ。じゃあな」

⋮

エドガー「おぉ，いいところへ来たな。この部屋のありさまを見たまえ」

マドル「はぁ，ずいぶんと小さいですね。Oh!X編集部より狭いなぁ」

エドガー「そうじゃなくて，この部屋の荒れ方だよ。私のいない間に誰かが入り込んで荒らしたに違いない。これは泥棒なんかじゃない。きっとバレスタイン城のやつらがやったんだ」

マドル「バレスタインといえば，今年はチョコレート1枚ももらえなかったなぁ」

エドガー「(無視して) 君がティグレーの採石場で手に入れた彫像があったろう。奴らはあれを私が持っていると思っているらしい」

マドル「義理チョコさえもらえなかったとは我ながら情けない……（まだ言ってる）」

エドガー「(さらに無視して) 君のような剣士でもいないよりいたほうがましだ（ひどい言われよう)。我々の仲間になってくれないか？」

マドル「だいたい日本だけなんだよな。チョコレートを渡しっこするのは（まだ言ってる)」

エドガー「わかったよ，私が悪かったよ。はい，それはバレンタイン，私の言ってるのはバレスタイン城」

マドル「いいですよ。仲間になりましょう」

エドガー「君と話していると，どっと疲れがくるよ」

マドル「みんなそう言います」

やってきたのは最初の冒険の舞台となったティグレー採石場。とはいっても最初の冒険では扉が閉まっていて行けなかった場所だ。エドガーさんの話では3つ目の彫像がここにあるという。

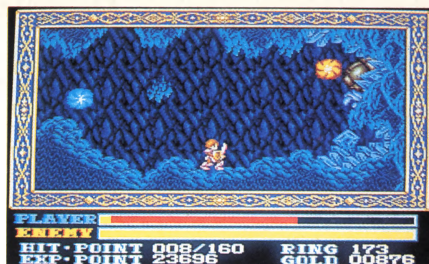
ここの敵キャラは，人間の血まで吸う吸血植物ローバルや，近づくと伸びる腕でパンチを打ってくるゲルド君など。ローバルは，触手の届かないところから剣を振れば一撃で倒せる。ただし，このローバルは意表をついたところに生えている場合がある，気をつけよう。ゲルド君は，ジャンプしながらジョイスティックを下に入れながら剣を振って下突きの状態にし，その頭上に着地するような感じで攻撃すればOK。レベルが足りないと一撃で倒せないかもしれないが，そういうアドル君達はレベルを上げるか，先月号で小鬼のラデル君を倒した方法（下突きの状態で剣の先を敵の頭に刺さるように飛越すのを繰り返す）を使うといいだろう。

おや，宝箱だ。中身は……。あれ，これを欲しがっていた人がいたっけ。誰だったかなぁ。ま，いいや。町に帰ってからゆっくり考えよう。

あ，行き止まりだ。おかしいなあ。これ以上先に行けないな。ここに来るまでに見落としたところはなかったかな。引き返そう。おやっ，ここは！　そうかイースIやIIではなかったアクション「××××ん×ん」を使って行くのか，なるほど。この先には敵の正体が明らかになるイベントがあります。わくわく。

さて，イベントが終わって下へ下へとしばらく進むとまたまた行き止まり。近くにはゲルド君が大勢，足元にはローバル。ここも少し頭を使わないと先に進めないぞ（とは言っても1秒で気づくとは思うが)。

しかし，こういった地形を利用した謎というのは実にすばらしい。「なになにを持っていないと先へ進めない」といった謎と違ってプレイヤー1人ひとりがその場で解決しなければいけない本当の意味での「謎」と言えるからだそういえば，今，大人気の「ダンジョンマスター」（ビクター音産)もこういった謎が中心だ。

無事に行き止まりの謎をクリアすると，3つ目の彫像を守っている生きた鉱石イスターシバ君との対決だ。装備を確認しよう。武器はブロードソード，防具はプレートメール，スモールシールドまたはラージシールド。えっ？　プレートメールは店で売ってなかったって？　そう，売ってません。ということは冒険中に手に入るってこと。持っていない人はティグレー採石場のどこかにあるから探しに戻ろう。レベルは8以上あればいいだろう。

恥ずかしがりやのイスターシバ君

おおっ，感動の再会！……でもないのか

イスターシバ君は岩石を降らせ水晶を自分に向かって吐いてくるが，これらを避けるのはさほど難しくはない。厄介なのは，くるくる回りながら飛び回る赤と青の火の玉だ。赤の火の玉はイスターシバ君本体の周りを回り，青の火の玉は少し離れたところを回っている。なにが厄介って，赤の火の玉に当たると青の火の玉の中にワープしてしまうのだ。よってなかなか本体を切りつけることができないのだ。これは困った。タイミングを見計らって攻めどきを狙うしかない。マドル様が考えた方法を図2に示しておくからどうしても倒せない人は参考にするといいぞ。

マドル=リステリン・イースIII冒険記〈中編〉でした。来月が最終回だ，もうひと月我慢して俺とつきあってちょうだい。

図2

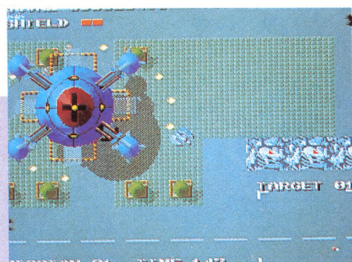
1．ここで水晶やガレキをよける
2．赤い火の玉がここに来た瞬間，ここにアドルをもってきて，斬りつける。しばらく赤い火の玉は来ない。

▶春だ！　今年は例年より雪が少なかったので1カ月も早くナナハンに乗ることができました。私の地方は当然冬になるとバイクに乗れません。
小山内　誠（31）X68000　青森県

●グラナダ

懐かしのタンクタイプアクションゲーム

Kunitsu Yoshio
国津 良男

ウルフ・チームの最新作は，いままでのビジュアルシーンビシバシのRPGやアドベンチャーとはうってかわって，懐かしい感じすらさせるタンクタイプアクションゲームだ。全8面で構成されている。

グラナダ：アルハンブラ宮殿で有名なジオン軍の月面基地……ちょっと違うな（広告の絵はGタンクみたいだったが）。

ふつうの人はグラナダといってもどんなゲームなのかイメージがわかないと思うし，「シューティングの夢」といわれれば，戦闘機ビュンビュンを想像してしまう。グラナダは正統派タンクアクションだ。いっそ，「戦車くん」とでもすればまだわかりやすかったのに。

男のバトル

さてもグラナダは男のゲームであった。戦争である。いかつい戦車に乗り込んで，縦横無尽に走り回り，目につく敵という敵すべてを，ただひたすら撃ちまくればよい。それだけだ。みみっちい戦略はいらない。当てになる仲間もいない。必要なのはサディスティックな闘争本能と8,800円だ。

さあ，いまこそ男の精神を呼び戻すのだ。同じ会社員なのに，男はしがないサラリーマンとさげすまれ，女は花のOLともてはやされる。悔しくはないか！　そしてなにより，ここは戦場だ。女の出る幕ではない。いざというときに頼りになるのは，男だ。「男は女の頭であり……女が男のために創造された」のだ（注1）。

俺は戦う。おまえは庭で雀でもながめながら，「ちひさきものはみなうつくし」とでもほざいてろ（注2）。

なぁんてね。冗談だってば。怒ったらごめんね，担当さん（←女性なのだ）。

システムの紹介

さてさて，自機戦車は2種類の武器を備えている。ひとつはバルカン砲，もうひとつは波動砲だ。これらは，あらゆる方向に撃つことができる。微妙すぎてよくわからないが，32方向以上サポートしているようだ。

バルカン砲より，波動砲のほうが威力があって，AB両トリガーを同時に押すと発射される。ただし，こちらは連射ができない。その隙に敵に撃たれてしまうこともままあって，つまりはめでたく一長一短ってわけだ。

でもって，どっちの武器にも弾数の制限がない。この豪快さは嬉しい。「あん，弾が切れちゃった。補給をしなくちゃ」だとか，「ボスキャラのため，弾の半分は残しておかにゃあならんぞ」，なんてセコビッチなことを考えていたのでは，アクションゲームの爽快感が損なわれてしまう。撃って撃って撃ちまくれ，なのだ。

走って走って走りまくれの戦場，バトルフィールドはおよそ8×8画面分。上下左右，斜めにもなめらかにスクロールする。が，いかんせん戦車が地上を走っているにすぎないので，スピード感に欠ける。こればかりは，どうしようもないだろう。タンクゲームの特徴ということで。ビルなどの障害物を取り去って，ハイスピードで移動できるようにすると，なにやら「サンダーフォース2」に似てきてしまいそうだし。

一方敵は，ザコキャラ，ターゲットキャラ，ボスキャラの3通りに分けられる，と今勝手に決めた。画面右下のレーダーは，ターゲットキャラの位置を示している。これを頼りにすべてのターゲットキャラを倒せば，ボスキャラがもったいぶってのご登場，という仕組みだ。ボスを倒せば面クリアの全8面である。なお，ザコはザコだ。箸にも棒にもレーダーにもかからない。あまり相手にする必要もないだろう。腐っても鯛，育ってもザコ。みにくいアヒルの子は，生まれたときから白鳥の未来が約束されていたのだ。なんて，怒ったらごめんね，PTAさん。

ミッション2

じゃ，実際のゲームを見ていくことにしようか。というわけで，ミッション1はすっ飛ばして，ミッション2，つまりは2面を見てみよう。そこは，宇宙ステーションのような近未来的なステージだ。眼下にも同様な世界が広がっていて，いわば2層構造になっている。なにやら神秘的で美しい。自機の移動につれて，下の世界もゆっくりとスクロールし，奥行きを感じさせてくれる。といっても，別に下のフロアと行き来できるわけではなく，落っこちたらそれまでだ。

まずは，レーダーを見て，全体の状況を把握しよう。ターゲットキャラの数は10ばかりか。道路から落ちないように注意しながら，ターゲットキャラの見える位置まで移動して，撃破，撃破だ。

X68000用　5"2HD版　8,800円
ウルフ・チーム　☎03(5273)4796

右下のレーダーでターゲットキャラが表示される

▶4月号の「ゲームシステム文学誌」はゲーム好きの私にとっては最高のOh！Xだった。
大久保　満（23）岩手県

このとき頭にくるのが，近くに敵がいるぞ，とレーダーが示しているのに，道路が途切れていて進むに進めない場合。ま，別に猫じゃあるまいし，ぐるっと遠回りすればいいのだが，タイムオーバー直前だったりすると，悔しい思いをすることになる。ちなみに，およそ4分でタイムオーバーになる。

で，ボスキャラだが，この面には2種類いる。初めに出てくるボスは，なんというか，パチンコ玉が寄り添ったような形をしていて，いってみればギラギラのボコボコだ。一見強そうだけど，波動砲を2発お見舞いしてやるだけで，簡単にやっつけられる。

次に出てくるボスは，ミサイルを鬼のように撃ってくるので，避けるに避けれない。が，後ろに回り込めば，そこは安全地帯だ。ミサイル発射口を破壊して，で，正面に戻って，あとは根性と反射神経で本体を攻撃しよう。

ミッション3

3面である。一転して，ここは荒れ地かそれとも砂漠か。どうやら夜中らしく，視界がぐんと狭くなる。カーレースのトンネル，イースの洞窟だ。

しばらく進んでいると，小さな青い物体が見つかると思う。これはオプションだから，必ず取っておこう。自機の頭にぴったりとくっついて，なにやらじゃまっけなのだが，AB両トリガーを押すと，敵めがけてミサイルを放ってくれる強力なパワーアップアイテムなのだ。

なお，この面にある岩は固くて固くてどうしようもないが，破壊できないわけではない。一応書いておく。

で，ボスキャラだが，これもミサイルを散発する。後ろに回り込もうと思っても，ボスがグリンと回転して，何の解決にもならない。地道によけながら，持久戦に持ち込もう。

＊　＊

こんなローラーにひかれたらぐちゃぐちゃや！

3面はこんなにも視野が狭いのだ

戦いはまだまだ終わらない。4面，5面……そして8面のあとのエンディングを見るまで。たとえ終わったとしても，友人とハイスコアを競う，という遊び方もできる。何といっても，スコアは億の桁まで用意されているのだ。そして，敵のミサイルを破壊して得られる得点は，たったの6点にすぎない。気が遠くなりそう。

操作性について

自機を動かしていて，ひとつムズいのが，砲台のみの回転ができないという点。敵に向かって弾を撃とうと思ったら，敵の方に突進しなければならないというのは，どうにも不具合だ。古くは「タイムパイロット」なんかもそうだった。

それでも，Aボタンを押しながらスティックを動かすと，同じ方向を向いたまま平行移動するという技が使えるので，いくぶんましだが，いろんな場合を想定してみると，Aボタンを押しながらスティックを動かすと，自分の進んでいる方向の反対側に砲台が向く，としたほうが使えたかもしれない。

2年ばかり前だったか，アーケードゲームに「アサルト」というのがあった。このゲームと同類のタンクゲームなのだが，これは，2本のスティックを用意することで，このへんの問題を解決していた。右スティックを前に，左スティックを後ろにすると，左回転，なんてふうに。が，多くの人は，その操作に慣れることができずに去っていったという話。難しいもんだ。

なかなか入りくんででめんどーだったりする

おわりに

唐突に思い出しちゃったけど，そういえばこのゲーム，ウルフ・チームのくせに(べつに悪い意味でいっているわけではないので誤解しないように！)，いつものノリの挿し絵，頻繁なビジュアルシーンがなかったぞ。この急な路線変更，うーん，渋い，渋すぎる。でも，個人的にはまったく無意味でもいいから，突然現れては消えてゆく，かわいいお嬢さんたちが欲しかったかなぁ。もちろんあくまでも個人的には，だけどね。

ま，いないものは何をいってもしかたがない。とりあえず戦いも終わったことだし，今夜はTOMOROSE（注3）で買ったハーブティーでも飲んで，のりピーちゃん（注4）のぬいぐるみと一緒におねむするとしよう。戦いすんで日が暮れて。しばしの戦士の休息だ。

（注1）新約聖書・コリント書第1-11章より
（注2）清少納言の枕草子ね
（注3）西村知美のお店。原宿竹下通りにある。かなり小さい店だ
（注4）さかいのりこデザイン。NORI・P・HOUSEは恵比寿1-1-6や竹下通りなどにある

ところで総評

ただ撃つだけ。単調。これが素直な感想だ。確かによくできてはいる。グラフィックは美しいし，マウスでも遊べるようにと気を使っている。しかし，タンクゲームであるということ自体が欠点を負ってしまっているような気がする。シミュレーションではなくアクションゲームなのだ。もっとスピード感がほしい。画面が埋まるほどの敵弾を，超高速で見切りよけきる，全身全霊を傾けて集中する，といった緊張感がほしい。よくアーケードゲームで見かけるような，人間が銃を抱えて歩き回るってゲームも同様だ。個人的な好みに過ぎないのかもしれないが，この手は私をダルな気分にさせてしまう。

どうせ迫力に欠けてしまうのなら，ピンクを基調にしたおもいっきりかわいいのにするとか，某ランデブーのように単なるじらしのために使うとかいったところが妥当なセンかもしれない。そんなわけで，私は「のりピーちゃんの大冒険」をいつまでも待つのであった。ポプコムさんが作らないかな。

グラフィック………………7
操作性………………………7
重厚長大度…………………8
ハマリ度……………………5
コストパフォーマンス……6
サウンド……………………未完成

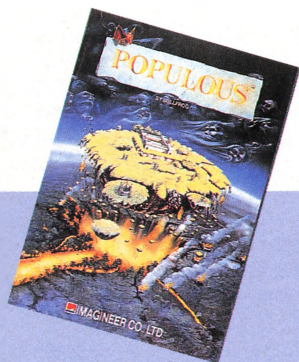
＊これは開発途中のサンプル版をプレイした結果を書いています。

▶ジョークを考えましたが，書けません……。つまりX68000をX10001101001…という感じにして「X1はX1だ」と言いたかったんです。68000を2進数で表すとなんになるのだろうか？
寺沢　靖雄（16）X1turbo　岩手県

●ポピュラス

神と悪魔の旗の下

Nakano Shuichi
中野 修一

文明，進化，そして聖戦。土地を築き，城を作り，人を増やし，さらに土地を開く。神の力と神の力が火花を散らす。天地創造をテーマにして，世界的に話題を呼んだリアルタイムシミュレーションゲームがX68000に登場だ。

X68000用　5"2HD版　9,800円(税別)
イマジニア　☎03(343)8911

「アーマゲドンの戦士を探しています。当方，創造主にして邪悪なる者……」

ポピュラスはイギリス産の天地創造シミュレーションゲーム（？）だ。ゲームには違う神を信仰する2つの種族が登場する。要するに神様になって自分の種族を生き残らせるのが目的だ。そしてゲームはどちらかの種族が全滅するまで続けられる。

神の右手

神様はどんな力を持つかというと，まず土地の造成。ゲーム中のほとんどは土地の造成に費やされるといってよい。では土地の造成がどんな意味を持っているのか？

ゲームを開始すると人々は地面の平らな部分を探して家を建てる。周りの土地が平坦ならさらに大きな家を建てる。そして，ついには城を作る。これを助けるわけだ。

大きな家ほど人の生産力が高く，またパワーによって画面上のサイコフレームと呼ばれるゲージがより速く上がっていく。サイコフレームは神様の使えるパワーの限界を表すもので，これは当然使うと減る。だから，人口や建物などは多いほうがいい。ゆえに，土地の造成が重要なのだ。ほかにもいろいろ技があって，土地の造成はポピュラスの基本であり，同時に奥義でもある。

ほかには地震，洪水，火山の造山，騎士を作るなど攻撃的奇跡も引き起こすことができる。まあ，これらは「そのもの」の動作なので特に説明もいらないだろう。

このような神の業を駆使していくわけだが，なぜか，ポピュラスが立ち上がるとプレイヤーはマウスを持った悪魔になる。

＊　＊　＊

ひと言いっておくと，このゲームはかなり危険なゲームである。いやいや，ゲームの性格とか，人間のダークサイドを露呈するとかいうのはたいした問題ではない。一度始めるとなかなかやめられないタイプのゲームなのだ。気がつくと夜が明けていることもしばしば。

といっても，延々と続くわけではない。マップひとつにつき30分から1時間で勝負がつく（途中のセーブも可能）。シミュレーションだと思えば短い部類だろう。実に手軽だ。だから，ついつい気軽に手が伸びる。最初は勝って当然，マニュアルを見なくてもだいたいの操作はできるし，土地を造成しているだけでも勝てるはずだ。プレイヤーは存分に善き神を演じることができる。民の発展に心を配り，無謀な侵略者を「痴れ者！」とばかりに洪水で一掃することもできれば，圧倒的なパワーで敵をいたぶることもできる。気分は「女王様とお呼び！」だ。ときどき怪物や魔法使いなどの変なものも出て邪魔してくれるがよくわからないので気にしないことにする。

だが，こちらも慣れた頃になると敵は次第に手加減をしてこなくなる。だんだん苦しい戦いが多くなるわけだ。よって勝ち進むと，さらに苦しいであろう次のマップが非常に気になる。対策を練るため，つい，設定を確認したくなる。さらにマップを覗いてしまうと「ちょっと小手調べ」……，見事にハマッてしまったわけだ。

そして，負けたときは……。

悪魔の左手

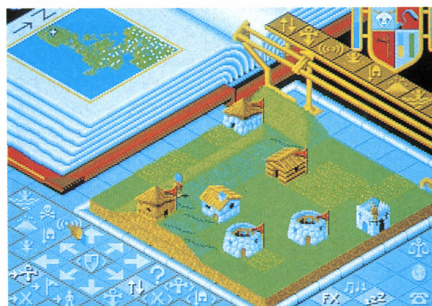
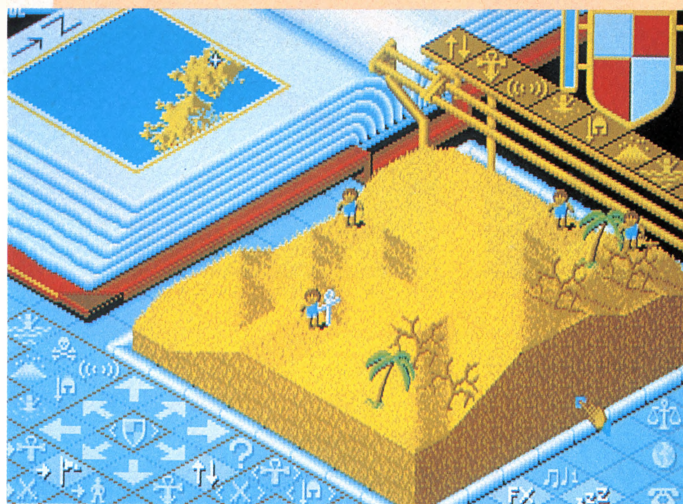
「ツブセ，壊セ，破壊セヨ……」

このゲームのもうひとつのポイントは「騎士」を作ることだ。

ポピュラスでは民族はいつのまにか増え，人と人を合体させることでより強力な民族となっていく。まるでデーモン族のような奴らだ。この合体を民衆のリーダーに集中することで傑出したパワーを持つ人材を作り上げることができる。そして，リーダー

まだまだ平和な世界

初めてハマッたのがここ

▶僕と友人は同じ高校に進むことになった。しかし，彼はX68000PROを買ってもらうそうだ。うらやましい……。
庄子　真海（15）X1turboZ　宮城県

全体図をクリックするとその周辺がズームアップ。サイコフレームが上がればマップスクロールアイコン左上のハデな奇跡が使える。下は集合，戦闘，開拓，合体のモードを指定するもの。「〈 〉」マークのアイコンは戦闘場面や騎士，リーダーをズームアップする

を完全武装させ人格改造すると騎士ができる。騎士は殺戮マシンという言葉以外では表すことはできない。圧倒的な力で敵を無差別に殺し，家を焼き払う。騎士の通ったあとには荒涼とした廃虚しか残らない。まさに侵略すること火の如し。

序盤ではシナリオの設定により相手は騎士を作れない。騎士はこちらの一方的な攻撃手段だ。しかし70面くらいになると，敵もどんどん騎士を作ってくる。当分は水攻めでかわす。私は悪い神様だなぁと思い始めるのがこのあたり。それでも敵方の騎士の脅威は十分に味わえるはずだ。

81面，私が初めて負けたシナリオだ。それまでは息絶え絶えなときもあったが，勝ち続けることができた。

勝てない。ここでポピュラスは別のゲームに変貌した。

相手の繁殖率が高く騎士の群れが襲ってくる。こちらは少人数，増えない，勢力拡散……。善き神のままではなすすべもなく嬲り殺しにあってしまう。で，そろそろ開き直ることが必要になる。所詮，血塗られた道である。

実際，負ける場合の終盤戦は悲惨だ。強力な敵の軍勢の前になすすべもなく斃れる民の姿をじっと見守らなければならない。無力感に打ちひしがれることになる。

RPGにしろシューティングゲームにしろ，プレイヤーは主人公と同一化することが多い。無茶をして死ぬのも自分ひとりだから責任は軽い。しかし，ポピュラスで死んでいくのはプレイヤー自身ではないのだ。弱き神のもとでは民衆は蹂躙され虐殺される運命にしかない。

やがて悔恨と憤りがふつふつと湧きあがり，神の業から悪魔の業の修得に励むことになる。

復讐戦

「そなたたちの最期しかと見届けた！ 我が復讐をバルハラにて見守るがよい！」

当面は敵に騎士を作られることがもっとも恐ろしいことだ。これを阻止するには……と考えていくと結局は一般的な戦略に落ち着くことになる。

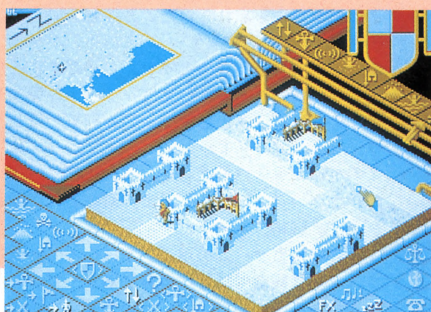
敵陣の中央を制覇し，敵を分断する。これで兵力の集中を防ぐことができる。強力な騎士を作るためにはなによりも1カ所への集中が必要なのだ。

敵を殲滅しつつ自分の勢力を伸ばすという基本を忠実に守りつつ，敵の攻撃圏外では広大な植民地を経営する。地上げは基本の基本だ。さらに，ありとあらゆる手(やり込むといくらでも悪魔の業を覚える）で妨害工作をする。いやがらせのつもりでも敵に有利になる場合があるので注意したい。洪水などの一発逆転の大技はそうそう都合よく使えないので期待はせず，無駄な用心もしないことにする。

あとはいかに効率よく人を増やし，強い民を作るかという神の業(これも奥が深い)を存分に揮う。これで勝てなければ，地形や条件をよく考えて新しい戦略を練ることになる。全500面だからまだまだ序の口。勝つ手はいくらもある。

結論，「善き神は強き神である」。

強き神になるにはどうすればいいか？これにはプレイヤー自身が強くなり，適切な戦略を迅速に実行する以外に道はない。マウスの誤操作は致命的でもあり，ときに新たな戦略を開いてくれることもある。ゲーム前のパラメータ表は見逃すな。

ちょっと苦しいが……

復讐ははたされた

総評

ちょっと見るとゲーム画面が小さいのがやや気になる。グラフィックはPC-9801でもまったく同程度。しかしサンプリングの効果音は絶妙。BGMも飽きがこない。

天地創造，多くの人が一度は夢見るゲームかもしれない。ふつうの人はちょっと考えて挫折する。確かボードゲームで文明を作るというものがあったが，それがリアルタイムに進行してしまうのだ。ルールが異様に簡単で，誰でもすぐにゲームに入れる。そうか複雑な事象はこうモデル化するのか，という模範解答のようだ。

アイデア一発のゲームではなく全体の完成度が非常に高い。全世界700万台の AMIGA ともなると，ヒットゲームのレベルも違うということか。ダンジョンマスターやシムシティも AMIGA が元祖だしなあ。ゲーム中はすべて英語表示なので雰囲気を壊さずにすむ半面，少々わかりにくいという人も出てくるかもしれない。

とにかくこのゲームに関しては世間の噂を鵜呑みにしても間違いはない。脱帽。 (S.N.)

操作性……………6
対戦……………10
アイデア…………9
サウンド…………8
グラフィック……8
自然神論…………9

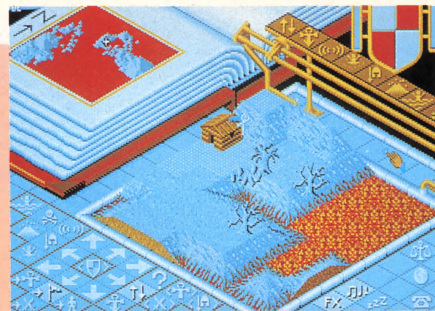
▶アサヒのCMで住田潮という名前を見て「おおー，私と同じ名前じゃないか」などと思ってしまった。以前，Oh！Xでも潮という名前を見たような気がするが，この名前は全国に何人ぐらいいるのだろうか。
山本　潮（20）X1turbo　宮城県

悪いことをしてなにが悪い?

「正義が力ではない，力こそ正義だ！」

さて，こういった人間のダークサイドを剥き出しにしてゲームしている姿は傍で見ている人には心地よいものではない。しかしピコピコとした一見可愛いキャラクターの動きに惑わされていると弱き神で終わってしまう。もっと平和にゲームしたいという人はシムシティを待ちなさい。

昔，徳を高めるのが目的というあるゲームがあった。徳ってなんだ？　ゲームを見ても納得できなかった。モンスターを殺し財宝を略奪する。それがおまえのいう正義なのか？　個人的な意見で申し訳ないが，私は正義という言葉が嫌いだ。「正義」という言葉が使われる場合，たいてい逆の行為がされている。そういえば某所のウイルスも「正義のため」だったなぁ。

本来，この世には法も秩序もない。あるのは「力」関係だけだ。世界有数の軍事力を持っていても，これは自衛のために必要なものだから軍事力ではない。守ってくれる者はなにもないから自分の身は自分で守る。これが原則だ。当然，大国が原子力空母や核ミサイルを装備するのもすべて自衛のためである。そして，ポピュラスでは身を守る最大の方法は自分以外を全滅させることだ。これは理にかなったことである。昔，国際政治を専攻した私がいうんだから間違いない(？)。

氷河の面はなかなか人が増えない

溶岩の海。でも落ちても平気？

そして対戦モード

そもそも平和なゲームでさえ対戦モードでは醜い争いが繰り広げられる。対戦テトリスでさえ人間関係を壊すことがあるかもしれない。そして，あらゆるゲームで対戦モードは面白い。人間対人間の戦いは理屈を超えた面白さを持っている。

ポピュラスはRS-232Cでマシンを接続することで（またはモデムで）対戦モードが可能だ。ポピュラスでは必然的に数々のいやがらせを対戦相手に行うことになる。そう，まるで悪魔のように。

また，メッセージアイコンでいつでも相手にメッセージを送ることができる。どんなときにどんなメッセージを送りたくなるかは容易に想像できるだろう。ああ，なんて恐ろしいゲームだ……。

ポピュラスによせて

先月号で“娯楽道具”としてのゲームと“娯楽メディア”としてのゲームという結論に辿りついた。しかし，ポピュラスを見てひとつ，大事なジャンルを**忘れていた**ことに気づいたのだ。それは，“箱庭観賞型ゲーム”である。パソコンのなかにあるひとつの世界を，ときどき手を加えながら観賞する。たとえば，ペットを愛でるようにである。この形式には“Ａ列車で行こう”や，AppleⅡの“リトルコンピュータピープル”。このポピュラスや最近有名な“シムシティ”，“パピーラブ”。コンピュータ同士で対戦しているときの“大戦略”もそうだ。

箱庭観賞型ゲームのポイントはいくつかあって，それは**箱庭**であり，**観賞**であり，**リアルタイム**である。今回は箱庭について考えてみよう。

まず，究極なことをいうぞ。**すべてのコンピュータゲームはコンピュータという閉じられた宇宙に構築された箱庭**である。だから，ただ箱庭型ゲームといっただけではいけない。箱庭を考えるとき重要なのは，誰が箱庭を作ったかである。箱庭の創造主だ。

娯楽メディア・娯楽道具型ゲームにおける箱庭の創造主はゲームデザイナーだが、**箱庭型ゲームにおける創造主はプレイヤー**なのだ。ゲームデザイナーが提供するのは箱庭の箱とそこへ並べるアイテム，そして箱庭世界を動かすアルゴリズムだけなのである。これが大事である。

箱庭といえば思い出すのが**箱庭療法**。心理療法の一種の芸術療法や表現療法の一種で，心理療法というのは，「すいません，うちの息子がおたくになってしまったんですが」とカウンセラーさんのところに相談に行くと，カウンセラーがそのクライアント(患者とはいわないみたい)を社会復帰させようとあれこれするわけで，そんなとき使われる手法のことだ。心理学とか精神分析学なんかの分野。で，箱庭療法というのはユング派のD.カルフさんが始めた方法で，何をするかというと，箱の中に砂とさまざまなアイテムを並べて箱庭を作るだけなのである。箱の大きさは内側が57×72×７(cm)と決まっていて，他に治療者は湿った砂と乾いた砂の入った箱，それから箱に並べる家だとか車だとかウルトラマンだとかのミニチュアアイテムを用意するのだ。

箱庭療法というのは日本で特にさかんであって，なんでかというと，日本人は言葉で何かを表現するのが苦手だったりするから。欧米人はすぐ論理的な解釈を欲しがったりしたりするけれど，日本人は自分のイメージや状態をうまく言葉で表現できない。カウンセラーのところへ通うような人はなおさらだ。だから，とっつきやすくて自分の抱えているイメージを出しやすい箱庭療法がいいのである。実際に作られた箱庭を写真で見たけれど，ずいぶん（特にアブない人の作った箱庭が）面白かった。

何がいいたいかというと，箱庭というのは作った人の心がはっきりと現れるのだ。当人がどう頑張っても，その人自身が表現されてしまうのだ。ねじ式よりもうひとりの自分と出会えるのだ。で，余談だけど箱庭療法が面白いのは，フロイト式夢判断のように，アイテムから解釈しないこと（象徴解釈をしない)。箱庭を作ること自体がクライアントにとって重要な表現となり，治療者はクライアントの持っているイメージを理解するのが大事なのだから。

で，ゲームにおける**箱庭度**というものを考えた。プレイヤーのイメージをどれだけゲーム世界に反映できるかである。たとえばポピュラスの箱庭度はあまり広くない。作って壊すことにしかプレイヤーの意識が集中しないからだ。せいぜいその人の破壊衝動度がわかるくらい。ネチネチと相手をいじめるか，ひとおもいに殺ってしまうか。自分の大地を大切に育てることに終始するか，相手の殺戮に終始するか，だ。しかし，箱庭度が狭い分，露骨であって面白いともいえるので欠点ではない。

Ａ列車で行こうは一見箱庭度が高そうだけれど，人の増えやすいところと増えにくいところがあって，特に第２作目以降はパズルゲームの色彩が濃くなってしまった。残念。

そんでもって，期待されるのがシムシティ。都市と箱庭のつながりは意外と深いのだ。都市と箱庭について書かれた文章もあるので，今度機会があれば（とりあえず，シムシティのX68000版が発売されてからだな。イマジニアさん，ポピュラスに飽きる頃には出してくださいネ）紹介しよう。

箱庭型ゲームは日本人によくマッチした形式だと思う。たとえば**庭園**であり，たとえば**華道**であり，たとえば**わび・さび**であり，たとえば**盆栽**である。**日本人は昔から限られた空間に自らの宇宙を凝縮するのが好き**だったのだ。そういったところを念頭に置いて現代風にパソコンでうまく表現できれば，なかなか面白いゲームが出来上がるに違いない。盆栽や華道をやろうとは思わないけど，自分の街や世界を作っていく作業はとても面白いから。

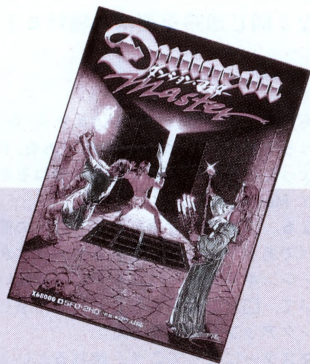
以上。大陸のフカン図や写真を見ると，すぐどこを平地にしようかなどと考えてしまうポピュラスな吉田幸一でした。　(K.Y.)

▶掲示板の「バックナンバー」の欄に，私が持っている号が出ているのを見ると，「むふっ！」と嬉しくなってしまう。でもそんな私も初めて買った号のBACK ISSUESを見るたびに「もっと買っておけばよかった！」と後悔する。　金原　真也（21）X1F　宮城県

●ダンジョンマスター

ダンジョンは深い あまりにも

Ogikubo Kei
荻窪　圭

3カ月間にわたってお届けしたこのダンジョンマスター物語も，今回がいよいよ最終回とあいなりました。でも，ここで書いたことはほんの一部でしかありません。なんといってもこのゲームは奥が深いのですから……。

X68000用　5″2HD版2枚組　9,800円(税別)
ビクター音楽産業　☎03(423)7901

なんか，先月号の自分の記事を読んで愕然とした。私は攻略法を書きたかったんだろうか。攻略法ばかりの巷のゲームレビューに逆上してこの仕事を始めたのではなかったか。これでは堕落ではないか。

というわけで，前号の予告は無視して，それでもせっかくだからダンジョンマスター物語は続けるのである。

なお，今月が最終階(じゃなくて最終回)である。ダンジョンマスターの魅力をあますところなく伝えられたかどうかはどうでもいい。なぜなら，ダンジョンマスターはプレイヤーの数だけエピソードがあるからである。途中で落としてきたアイテムや開かなかった扉の数だけ物語が残るからである。旅立てなかった勇者の数だけ伝説があるからである。勇者の腹におさまったネズミのもも肉の数だけ物語があるからである。ダンジョンマスターが与えるのは，そんな物語の本線だけにすぎない。

ふと気づいた。ダンジョンマスターの醍醐味は，マッピングをしないことにあるのではないか。マッピングをしてはいけない。視線を方眼紙なりなんなりに移すたびに，気が散るからだ。私がいるのはダンジョンであり，決して炬燵の前ではない。

格子の向こうに宝箱がある。あれが欲しい。格子の右手にレバー。そこで僅かな警告を意識の底から感じる。私は疲れていたのかもしれない。無批判に引かれたレバーの向こう，閉まったままの鉄格子の向こうに箱はなく，ただ床の穴だけが笑っている。憤慨と後悔。5分間の無駄な努力のあと，階段の下の小部屋で落ちた箱を発見した。開かなかった格子は，のち，裏側から開けられることとなる。

歩く前と通り過ぎた後。どれだけダンジョンの形を変えたか。ドアを開け，壁を開いた数。それが知恵の証明。

かにさんかにさん，どこ行くの。壁の小さなボタンを探し，隠れた部屋へ参ります。重いけど硬い鎧や，魔法の剣を探しに参ります。

巨大な目。宙に浮く金色の大目玉。魔法を操り，火の玉を放つ。前方から近づく火の玉をかわし，近づいて切りかかる。危ない。2歩下がり，敵の魔法は避けれるよう，こちらの魔法はあたるよう願う。呪文を唱え，ゴスモッグが火の玉を放つ。ふっと大目玉が消える。脇道へ避けたのだ。無駄な

自殺の仕方

ダンジョンマスターで遊んでいると，実に頻繁にキャラクターは死にます。ちょっとした心の油断をついて，死にます。大事で手に入れがたいアイテムを入手した直後や，ヴィーの祭壇の近く，何時間もセーブせずに進んでしまったときならば迷わず“遺骨に新たな生命を”吹き込むことでしょう。しかし，どこに祭壇があるかわからない深いところで戦士がひとり死んでしまった，とか道に迷ってうろうろしていたら，横から来た敵に僧侶が殺られてしまったときなどは，わざわざ復活させるのも面倒です。

そんなとき，パーティの残ったメンバーは自殺を図りたくなります。そこで，自殺あれこれを紹介しましょう。

・壁に頭をぶつける：これはキャラクターが共に弱っていたり，まだ低いレベルのときだけに通用するドロくさい方法です。一度に2人までしか死ねません。確実ですが時間がかかるので，高い精神力が必要となります。

・穴に落ちる：俗にいう飛び降り自殺というヤツです。弱ってない限り即死は難しいため，何度も飛び降りることになります。これも原始的な方法ですが，穴が近くにないとできません。

・餓死する：敵がいないところで，眠ったまま放置しておきます。寝ている間は，通常の何倍もの速度で時がたつので，腹は減り，やがて餓死します。プレイヤーの操作が不必要なため楽ですが，時間がかかります。眠ったままの死は，直接手を下したくないプレイヤーの逃避姿勢と密着します。

・走りまくってスタミナ切れで死ぬ：こんな馬鹿なことをする被虐的なプレイヤーに使われるキャラクターは可哀相です。

・自分の放った矢に当たる：これはなかなか高等技術。ワープゾーンをうまく使うと，前に放った矢が右から飛んできて当たる，といったこともおきます。うまくいくと楽しいけど，自殺にそこまで手間をかけるものでしょうか。

・モンスターに身をまかせる：勝手にこうなってしまうことも多いのですが，自ら，モンスターの前で無為に過ごすというのもなかなか忍耐のいる仕事。昔むかし，自らの肉体を鳥につつかせる苦行をした僧を思い出します。見ていると，つい手を出したくなるのを我慢する精神力はたいしたもの。

・壁に向かって，火の玉を放つ：私の愛用する自殺法です。強力な魔法使いがいると，たいてい2発もあれば全滅です。焼身自殺というやつですな。

・毒の雲を作り，中に飛び込む：うーん，苦しそう。忍の一字。毒を食らわば皿までどーぞ，てなもんです。毒の中で悶えながら死んでいく勇者の姿が哀れを誘います。

・火の玉を放ち，それを追い掛けていって，爆発に巻き込まれる：アハハハ，馬鹿馬鹿しい，アハハハハハハ。

どれにしろ，死の瞬間の叫び声だけは聞きたくはないものです。

▶「(で) のぱーてぃハンズ」のような記事をもーっと掲載して，X68000でみんながソフトを作れるように指導願いたい。勉強不足ももっともであるが，まず手本があるとないとでは，初心者は覚えるスピードが違うと思うのですが……。あと，ソフトサービスなども行ってほしい。
斎藤　雅文(24) X68000 秋田県

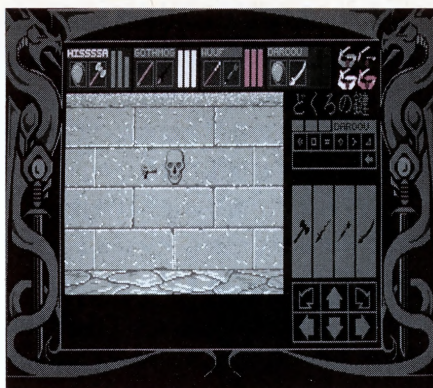
マナに悪態をつく私の隙を狙った大目玉の火の玉。構わず突進し，逆上したヒッッッサが斧で切りかかる。ああ，敵もまた我らと同様に，やはり避けることもあるのだ。

ラーの鍵計3つ。炎の杖にまた近づく。どんな鍵を持っている？　鍵係のゴスモッグ君。ああ，それでは炎の杖にはまだ遠い。でも，気をつけて行きなさい。ドアの向こうの緑の杖，それがいつしか手にする炎の杖。きびすを返すのはまだ早い。あのドアの向こう硬くて強い石人形。彼らを倒すと暗黒の剣があるでしょう。次に来るときにはきっと炎の杖は君らのものだ。

足下に注意，頭上に注意。床に溢れるスイッチ。それは向こうの壁から火の玉を発射するスイッチ。とても便利な侵入者撃退装置。冷静に。火傷はおっても，体の傷なら魔法で癒せる。

深いところなら，どのフロアにもあるどくろのマーク。どくろの鍵を挿せ。鼻の真ん中に挿せ。秘密の壁が開く。その向こうは最下階へ続く直通階段の踊り場。下へ行って体を癒すもよし。上へ戻って死者に魂を吹き込むもよし，水や置き去りにした食料を補給するもよし。たいてい下る階段の近くにある命を救う抜け道。そこへの道は

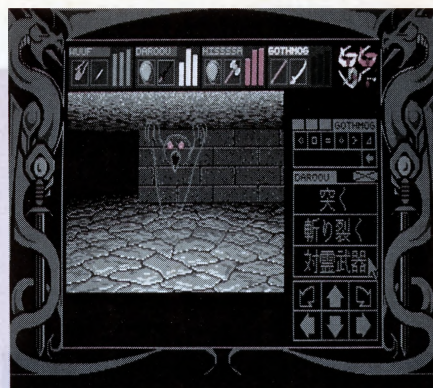
どくろのマークにはどくろの鍵を

うしろが透けてる……あ，ユーレイか

忘れるな。目印を置いていけ。

どくろの鍵を探せ。必ずどこかにある。

＊　＊　＊

常識的に考えてみよう。たとえば，十字架といえば，プリーストが首にかけるものではないか。

実をいうと，私はサソリ座だ。だからといって，サソリを見て異国の果てで旧知の友と会ったような感情を抱けとはいうな。あの金色に輝く巨大なサソリを見て恐怖を感じない人間があろうか。よるな，よるんじゃない。お前なんか嫌いだ。毒を注入するな，馬鹿野郎。

ついでにいうと，私は虫が苦手だ。たいていの虫は嫌いだが，そのなかでも特にクモが駄目なのだ。蜘蛛という字を見ただけで背筋が凍るのだ。ディズニーランドのホーンテッドマンションは大好きだけど，序盤に出てくる巨大なタランチュラだけは正視できないのだ。頼むから，蜘蛛だけはやめてくれ。しかし，ダンジョンマスターの巨大グモだけは安心。だって，あれは，蜘蛛というより，ウルトラセブンはベル星人の回に出てくる浮遊大陸の怪物（名前を忘れてしまった）を黄色に塗ったようなやつだからだ。だから，いくら強くても，突進していく勇気は萎えない。

目の前に2つのドア。どちらを選ぶ？　どちらでもどうぞ。前世の因縁か日頃の行いか，運の悪かった人はちいとばかり強い敵や大量の敵との出会いがあるだけだから。それもまた悪い因縁を断ち切るための自己鍛錬と思えばなんということはない。いま苦労してもあとで苦労しても朝三暮四だ。

君は自動歩道を知っているか。上に乗ると自動的に次のマスへと運んでくれる楽チンな歩道だ。しかし，そのスピードが尋常でないとしたらどうだろう。しかも，一度乗ってしまったら，死ぬまでぐるぐると終わりなく同じ通路を運ばれ続けるとしたら

マナ

マナ。魔法の素ですな。元の意味は何かな，と思ったのだが，編集部のスタッフには英語版を遊んだ人がいず，マナの綴りがわからない。

で，マナってなんだ？　となったとき，真っ先に候補に上がったのがキリスト教のマナ。綴りはMANNA。天から授かった食べ物の意味である。語源は旧訳聖書の出エジプト記。イスラエル人が荒野をさまよっていたとき，食べ物がなかった。それで，神が天からマナを降らせ，人々はそれを40年の間食べ続けたというわけ。マナにまつわる話はいろいろとあるけれど，とにかく，天から降ってくる食べ物で，その日のうちに食べないと，翌朝には虫がたかって食べれないというものであった。

でも，食べ物と魔法の素はどうもつながりがわからない，と，私は思ったわけで，仏教の世界でもマナという言葉があることを知った私はそちらを調べてみた。マナ識という7番目の識があって，これは思い量ることを指す。マナ識のマナはサンスクリット語でMANASと書く。考えることという意味。怪しいぞ。でも，サンスクリット語がポンと出てくるのも妙だ。さらにマーナというのもあって，日本語では慢（まん）。慢心の慢なわけで，これもまた変だ。

で，調べてみると，あったあった。1891年，コドリントン（変な名前）という人がメラネシアを調べていて発見した言葉だ。綴りもMANAと素直。で，そのメラネシアでいうマナはどういう意味かというと，「……あらゆる方法で善と悪に働き，それを所有し支配すれば最大の利益を得るような超自然の力（宗教学辞典）」，つまり，魔法の素，呪力の素のマナだ。あらゆる事物（人や動物など）にマナは宿り，ある樹のマナをほかの動物に移したり（転移性），マナが自発的にほかに移ったり（伝染性）する。文字どおりダンジョンマスターのマナとピッタシだ。魔法使いってのはマナを集めて使う能力を持った人のことだったのだ。ちなみに，そのコドリントンさんが1891年に書いた本から抜粋しよう。

「もし人が戦いに勝ったなら，……（中略）。それはたしかにかれを力づける精霊あるいは死んだ戦士のマナを得たからで，マナは首のまわりにつけた石の護符，腰帯につけた葉の房，左手の指に掛けた歯，あるいは自分に超自然的な助けを引きつけるに用いる特殊な形式の言葉などの中に包み込まれる」

参考になっただろうか。ダンジョンマスターでいうと，マナを上手に使いたければ，首の護符（ペンダント）や腰のお守りにまで気を配れ，っていうことだ。超自然な助けを……。言葉とは，文字どおり呪文のことで，マナは呪文のなかに包み込まれるのだ。うーん，奥が深い。

参考文献

[1]宗教学辞典　東京大学出版会
[2]新仏教辞典　誠信書房
[3]岩波仏教辞典
[4]食べものからみた聖書　日本基督教団出版局

マスター

先月号では，技術レベルはエキスパートで終わっていた。で，その先の話である。Expertになって，さらに実戦で修業を重ねると，

くく Master

になる。マスターである。マスターの前にあるくくであるが，これは魔法を唱えるときのパワーシンボルだ。くくはローで，一番低いやつ。続いて，口（ウム）になりといった感じで，ランクアップしていくのだ。

マスターになってこそ，誇れるというもの。

X 68000もExpertの次はMasterかなって思ったら，違ったね。

▶「ROGUE」をX1でやるのならCP/M版があります。タイニー版という気もしますが，それでもかなりハマります。　沼田　靖志（21）X1F/turboZ，MZ-2500　山形県

そして深部へ戻ろう

どうだろう。さあ，勇気を出して，永遠に回り続ける自動歩道から飛び降りるのだ。タイミングを誤って壁に頭をしたたかにぶつけても，臆してはいけない。うまく向こう側へ飛び降りれたものだけに，明日はやってくるのだ。神にばかりすがるな。天は自ら助くる者を助けるのだ。フラワーマン（BYボ・ガンボス）は“どこへいっても全部いいところ”であることを悟ったときに初めて現れるのだ。助けを求めるだけでは何も得られない。

＊　＊　＊

いま来た道を引き返せ。これほど的確で便利なメッセージがあるだろうか。

時計回り。これほど簡潔で人を陥れるメッセージがあるだろうか。どう歩けば何が現れるか。わからぬまま，格子の奥へ続く壁は開かれる。永遠に続くドーナッツ回廊でもやはり，天は自ら助くる者を助けるのである。神にすがるだけで感謝しない者に，現世利益だけを求める者に，真の姿は見えやしない。

視野を広げよ。つまりはそういうことだ。臆病者は追い詰め餌食にする。こういうことでもある。餌食にされたくなければ，足を地につけ，戦うことである。

二刀流の騎士が呪われた兜，呪われた鎧に身を固め切りかかってくる。火星人の幽霊のような奴が，火の玉を吐く。もう何をいうことがあろう。幽霊に斧で切り掛かっても無駄だ。呪われた騎士の鎧に慈悲を求めても無駄だ。

鉄の防具よりも黒鉄のほうがカッコいい。洒落てる。しかし，重い。それより白金(つまりプラチナだ）のほうが高価で軽い。私の友達にプラチナのネックレスを外さないヤツがいる。予想どおり，ヤクザみたいな見掛けのヤツだ。プラチナで身を固めるとは，つまり，どういうことだ？

以前通った，火の玉を発射する侵入者撃退装置。久しぶりに訪れたら，ネズミの死骸，つまりもも肉がいくつか転がっていた。我らを追おうとして，火の玉にやられたのだろう。しばし冥福を祈って腹におさめる。

＊　＊　＊

ここは深い，あまりにも。

たとえば，炎の玉を乱発する怪物。たとえば，床のいたるところで燃え上がる黒い炎の霊。たとえば，全身黒づくめの邪悪の香りする巨大な魔法使い。その場を逃げたとして，誰が責めよう。逃げるのは決して悪いことではない。悪いのは逃げ込んで，そこから出てこないことだ。観念しよう。安住の地はここにはない。

炎の杖。そこは偉大なる魔術師で杖とこのダンジョンの真の持ち主，グレイロードの研究室，実験室。さまざまな道具。フラスコ，魔法の石，虫眼鏡，呪文の巻物。さらに奥へ。魔法には欠かせない4元素。火，土，水，風。偉大なる錬金術。マナをこの手に。

グレイロードは慎重だった。巧妙に隠された壁の奥。秘術を伝え，炎の杖の力を記述した文書。

最後のラーの鍵。長い通路。石人形。炎の杖。

深部へと戻ろう。恐怖は克服できる。降りるのは簡単だ。杖の力は最下階でしか得られない。偉大なる魔術師が，そこへと至る隠れた道を残しておかないとでも思うか。私はそれに気づくだけの洞察を欠いていた。身体はアクティブに頭はダイナミックに心はスタティックに，だ。これが基本。

不気味なほどの静けさ。マウスの指が汗ばむのを止められない。身体を休め，マナを蓄える。

ここは深く，あまりにも深い。

ドラゴンがいる。かわせ。お前の相手をするにはまだ力が足りない。

炎の杖を燃やせ。そして，謎を解け。

つまりは，こういうことだ。どんな絶望のあとにも，未来だけは必ずやってくる。悪人の上にも善人の上にも。

うまいものを食って死ね

パン，林檎，チーズ，とうもろこし，もも肉，スクリーマーの肉，紫虫の肉，ドラゴンステーキ。ダンジョンに食べ物は欠かせない。

どれがおいしいか，つまり，どれがいちばん腹を満たすことができるか，っていえば，筆頭はドラゴンステーキだ。最下階のドラゴンを倒すと，10枚前後のドラゴンステーキになる。続いて（ドラゴンステーキなんてめったに食べられないから），もも肉であろう。命がけでネズミ狩りをするだけの価値はある（でもネズミの肉なんだよなぁ）。

どちらもないときは，チーズである。つまり，チーズともも肉は，おいしいものはいちばん最後に食べようの精神により，取っておこう，といっているのである。“もも肉を食うまでは死ねるかぁ”の精神である。紫虫の肉なんかは不気味な上にまずいので，さっさと食べるなり，道標にしてしまうなりしよう。

汝，隣人を愛せ

長く遊ぶにつれ，感じる。連れ添いは慎重に選ばなければならない。ただマナが多いとか，装備がいいとか，連れていくと有利そうだなどという打算で選んではいけない。

長い旅は気の合う，思い入れの持てるヤツを選ばないと，すくすくと成長してはくれないのだ。絶対，ひいきのキャラクターや思い入れの持てないキャラクターができ，両者の差は大きく開いてしまう。好きなヤツが死ぬと悲しいが，どーでもいいヤツが死んでも“こんなところで死ぬんじゃないっ”と怒るだけだ。私はヒッッッサとゴスモッグに思い入れを抱いてしまった。だから，他の2人はかなり強いにもかかわらず，つい彼らを多用してしまった。

どんなキャラクターも，強くなればなるほど性格が際立ってくる。こんなゲームは初めてだ。

総論

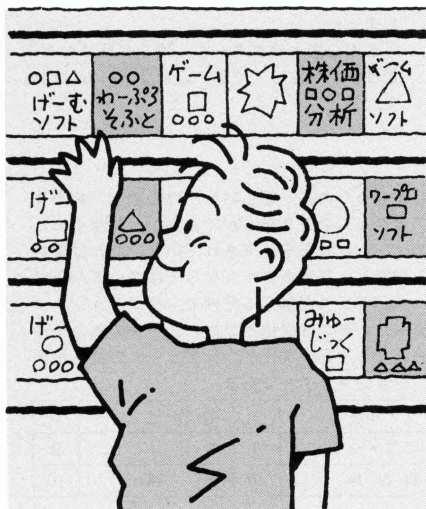
漫画が手塚治虫以来ずっとそうであったように，ゲームも映画の手法を学び，演出に取り入れようとしてきた。違いといえば手本だ。漫画は常に外国の優れた映画を見て近づこうとしたが，ゲームは漫画が得た手法を元に作られたアニメ映画を踏襲しようとした。そして，イースをはじめとするストーリー重視のエンターテイメントゲームが一世を風靡した。しかし，である。外国人はそんなこと構ってなかったりするのだ。このゲームはその代表である。プレイヤーを楽しませるための，新しいしかけの数々。あのリアルタイムのシステムができあがった時点でこのゲームは将来を約束されていた。あとは舞台を設計し，リアルタイムならではのパズルを付け，バランスを整えるだけで，ダンジョンマスターだ。アニメ文化とコンピュータの融合がイースなら，冒険文化とコンピュータの融合がダンジョンマスターである。

リアル度	9	操作性	8
アニメーション	9	スリル	9
叫び声	10	迷路	10

▶表紙も新しくなったことですし，スタッフの皆さんも気持ちを新たに頑張ってください。
半澤　崇志（16）X68000　福島県

AFTER REVIEW

AFTER REViEW

今月からAFTER REVIEWにタイトル変え。文字どおりゲームをやり込んだ感想をライター，読者問わず載せていきます。しかし，まだまだみんなのハガキが少なぁいっ！　もっとみんなのホンネが聞きたいぞ。もっと意欲的になってくれることを願う！

バブルボブル

▶移植がバッチリで面白い（Good）。
神奈川県・中島　正（20）

▶発売日の次の日に買った。スゴかったぜ！　移植はほとんど完ペキ!!
神奈川県・鈴木　康之（19）

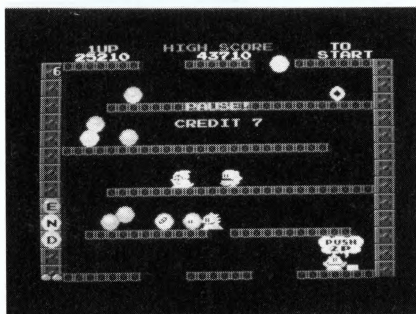
▶単純で面白いからよい。
京都府・福知　健（18）

▶ゲーセンでハマったゲームが家でそのまま遊べるのは，非常にうれしい。
兵庫県・稲山　直哉（19）

▶ゲームセンターで見たときには，正直いってそんなにやりたいと思わなかったんだけど，でもこうして自分の家でやってみると…ハマってしまったんだな，これが。純粋に面白いぜ！　あー，なんでゲーセンでもやらなかったんだろう。　千葉県・牛久　慎一（22）

▶アーケード版とほぼ同じくらいの雰囲気。
福岡県・日南　修一（17）

▶ファンタジーゾーンと同じようなできのよさ。　兵庫県・吉田　宅児（18）

▶バブルンとボブルンがかわいーから，好きっ！　敵キャラもかわいいんだけど，まだちょっとそこまでよく見るヒマがないの。ヘタだから……。あのお花畑を見られたときには，もう感動の嵐！　やっぱりゲームはこーいう女の子でもできるようなのがいいな。
東京都・高橋　美樹（17）

▶知っているとお得なテクニック
・「E」「X」「T」「E」「N」「D」の文字のうち3回同じものを取ると「杖」が出る。
・スコアの10桁と100桁の数字を同じにして敵をやっつけると，すでに吐いてあった泡もフルーツとなる。
・上から下へのワープを繰り返すとアイテムが出る。
・たくさん歩くと「靴」が出る。
・泡をたくさん割ったりたくさん吐いたりすると「あめ」が出る。
・すかるもんすたを30回くらい出すと，すかるのアイテムが出て流れ星が流れる。
（西川　善司）

今月送られてきたハガキのなかでも人気が高かったのが，このバブルボブル。ルールもやさしく，敵キャラ，自キャラともにかわいいせいもあって，なかなか好評のよう。いかにもタイトーのゲームらしい，といった感じがします。ハガキに書いてあった感想も「面白い」「単純明快」などといったのが多かったですね。

このゲームは，システムがどうの，グラフィックがどうの，といった面倒なことは一切考えずに，楽しむことに専念できるのがいいですね。敵キャラにやられても，ミサイルでドカン！　とやられたときほど腹も立ちませんし。フルーツやケーキも美味しそうで，なんとなく食べたくなってくるのは，私がいやしいからだけではないでしょう。コンティニューもあるし（ただし制限があるけどね），誰でも遊べるゲームに仕上がっているのがうれしいですね。まさによくできた移植作品の部類に入るでしょう。ぜひいちどプレイしてみてください。

X68000用　5″2HD版　7,200円
電波新聞社　☎03(445)6111

発売中のソフト

★Misty 4

Mistyシリーズの第4弾が発売。今回もシナリオ5本を収録。ユーザーの考えたシナリオに挑戦だ！　メモを片手に推理に頭をめぐらしてほしい。

X68000用　5″2HD版　5,000円
X1turbo用　5″2D版　5,000円
データウエスト　☎06(968)1236

★ギルガメッシュ・ソーサリアン

おなじみソーサリアンの追加シナリオ集。今度の「ギルガメッシュ・ソーサリアン」には“シュメール”，“ソドム”など古代文明を思い起こさせる5本のシナリオが収録されている。セレクテッド・ソーサリアンと違って順番に解いていくと全貌が明らかになるタイプなので，いままでとはまた違った楽しみが味わえるぞ。

X1turbo用　5″2D版　3,500円
ブラザー工業　☎052(824)2493

新作情報

★闇の血族

ノベルウェアに新たにミステリーシリーズが加わった。“A・D魅由”シリーズがそれで，第1弾がこの「闇の血族」だ。

主人公の魅由は新人アパレルデザイナー。ある日友達のモデルが変死し，続けて親友も殺されたことから，魅由は事件の追跡を始めるが……。アニメーションにのせて繰り広げられるミステリアスアドベンチャー，美少女名探偵の活躍に期待だ。

X68000用　5″2HD版　8,800円
システムサコム　☎03(635)7609

★クォース

コナミのシューティングパズルゲームがX68000でも遊べるようになった。キューブが接近してくるので，さらにキューブを打ち込んで四角形に整

▶最近のゲームって何人用とか書いてないのがある。袋とじの本と同じでパッケージを見て買うのが多い。その辺を考えてパッケージにも力を入れてほしい。この前やっとシリコンキーボードカバーを買ったのでX68000キーボードを死ぬほどたたいてます。
遠藤　英樹(17) X68000 福島県

サンダーブレード

▶ヘリコプターの浮遊感が非常によく出ている。　東京都・滝沢　邦明（22）

▶すごい，すごいぞ～！　さすがSPS & SHARP！　と思った。　埼玉県・加賀谷　匠（15）

▶やり始めると時間を忘れてしまい，気がついたら朝の4時だった。　静岡県・田中　宏典（16）

▶渋い。　栃木県・鹿又　健（20）

▶拡大・縮小がすごい。　茨城県・上遠野　雄一（18）

▶MIDI対応だから。　愛知県・林本　一成（16）

▶撃って撃って撃ちまくる壮快感が，たまらなくいい。　東京都・青木　学（18）

このサンダーブレードも人気が高いですねぇ。ゲーセンでも流行ってたし，そのあたりの影響もあるのかな。なんといっても，ヘリコプターの浮遊感覚がたまらなくいい！　それに2D面と3D面の境がスムーズで，違和感なく楽しめるのもいい。ただ，ポーズができないのはちと痛い。まあ，ゲーセンでプレイしてたら，トイレどこじゃないわけだしな。MIDI対応だし音楽もなかなかですぞ。

X68000用　5″2HD版3枚組　9,800円
シャープ　☎03(260)1161

ファーストクイーン

▶マルチキャラ，勝手に戦闘してくれるシーンは見もの。　大阪府・谷口　博一（23）

▶ボクはボコスカウォーズが好きですから。　大阪府・保田　周作（17）

▶「シルバーゴースト」を買ってみようか迷ったんですよねぇ。結局やらなかったけど……。懐かしさで1票。　神奈川県・中内　崇夫（21）

▶おもしろいよ！　長野県・須澤　加実（17）

うーん，わちゃわちゃと動き回るキャラクターたちがキモチいいっ（ん，なんか違うって？）。このゲームも，地味ながら着実にファンを集めつつあります。多人数対多人数で戦う姿は，戦争をしているにもかかわらず，ほんとにかわいらしい。個々のキャラクターにちゃんとパラメータがあるのも泣かせますね。大事にしなくちゃっ，て思いますもん。

しかし，です。やり込むとけっこう面白いんだけど，最初はなにがなんだかわかんないのがネック。ボコスカウォーズを知らなかったらマニュアルとにらめっこするか，30分くらい成り行きに任せるかしないといけなそう。いいゲームなんだけどなぁ。

X68000用　5″2HD版2枚組　8,800円
呉ソフトウェア　☎048(646)0660

アルガーナ

▶X1用の新作ソフトがこれしかないから。　静岡県・宮城　義和（26）

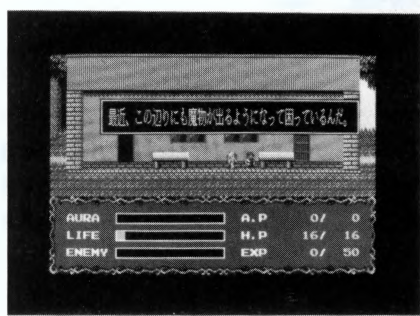
▶ノーマルX1で動くなかでは，いちばん新しいやつだ。　愛知県・加藤　富盛（17）

▶5重スクロール，漢字表示，PSGでのサウンドが気に入った。奈良県・松田　徹（14）

▶新，X1ユーザーの友！　石川県・米倉　博（16）

▶ひさびさにわかりやすく，すぐ解けてしまった。が，いまどき6,800円で売るわりには，エンディングもかっこよくていい。でも，ひとつ文句いわせてくれるなら，「なぜFM音源じゃないんだっ！」。　長野県・竹村　義彦（19）

「ソーサリアンに続くX1のRPG」との呼び声が高いこのアルガーナ。背景処理や漢字表示など細かい点まで注意を払ったところが，ユーザーの心をとらえたのかもしれません。なかなか雰囲気もいいのですが，どうしてもソーサリアンを思い起こさせてしまうのが残念です（ソーサリアンの3重スクロールに対抗して5重スクロールにしたらしい……）。でも，楽しめるのは確かですし，今後もM.N.Mには期待したいものです。

X1turbo用　5″2D版3枚組　6,800円
ブラザー工業　☎052(824)2943

えて消していくゲーム。まとめて消すと高得点だ。対戦モードや協力モードで楽しむこともできる。

X68000用　5″2HD版　6,800円
コナミ　☎03(262)9110

★ジェミニウイング

テクモのビデオゲーム「ジェミニウイング」が，システムサコムによってX68000に移植された。虫に覆われた大地を取り戻すというストーリーからわかるとおり，昆虫を意識した敵キャラはなかなかの迫力。移植のほうもMIDIやAD PCMに対応，縦画面モードも用意したという入魂の一作だ。

X68000用　5″2HD版2枚組　8,800円
システムサコム　☎03(635)7609

★サーク

マイクロキャビンの自信作，サークがX68000に登場だ。主人公は，ウェービス国のフェアレスという村に住む青年。しかし，彼はかつて怪物バドゥーを封印した，戦いの神デュエルの末えいだった。村に届いたバドゥーの復活の知らせを聞き，彼はバドゥー封印の旅に出る。「VRシステム」と呼ばれる自然感覚の映像表現は注目だぞ。

X68000用　5″2HD版3枚組　8,800円
マイクロキャビン　☎0593(51)6482

★LIFRAIM

LIFRAIMはパズルゲーム。アルガーナが評判のM.N.M.softwareの作品だ。ネズミ君が滑車を伝って，チーズをガールフレンドのところまで持って行くというゲームだ。明るいBGMを聴きながら，おもりや高さの調節の知恵を絞ってほしい。

X68000用　5″2HD版　6,200円
ブラザー工業　☎052(824)2493

★ブレード・オブ・ザ・グレート・エレメンツ

新規参入ソフトハウス“アミューズメント”のデビュー作。横スクロール型のシューティングゲームだ。「甦った精霊の力によって豹変した世界。この原因を知ろうと，精霊使いの主人公は南アスタリアを旅立った……」というようなストーリーだ。

従来の“アニメ調”とは一線を画す，クオリティの高いグラフィックは必見だ。発売は4月下旬の予定。

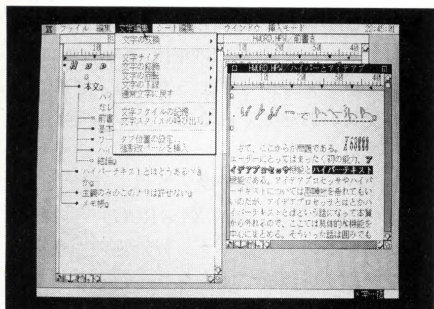
X68000用　5″2HD版2枚組　価格未定
アミューズメント　☎03(5396)3759

★サイクロンExpressα

サイクロンシリーズにポリゴン対応の新バージョンが登場。従来のプリミティブに加えて，任意の形状のポリゴンデータの処理が可能になった。速度の低下も，ボクセル分割によって防いでいる。画質の向上にも新技術が投入され，ジャギーの出ないマッピングができる。パソコンレベルを超えた画像作成を可能にする，強力なツールといえそうだ。またZ'sTRIPHONY DIGITAL CRAFT（ツアイト）のポリゴンデータもサイクロンExpressα上で再現できる。

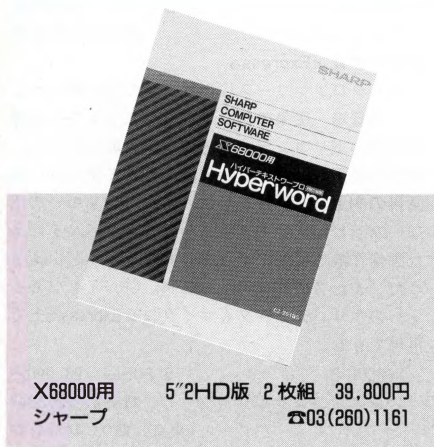
X68000用　5″2HD版　98,000円
アンス・コンサルタンツ　☎092(522)6347
（東京）☎03(447)4144

▶3年ぶりにOh!Xを手にした。うーむ，すっかり変わってしまった。ゲームの鬼，清水和人氏などの名前が見あたらない。ムッ？　MZ-700にスペースハリアーが発表されたのか！　くっ……，それは知らなんだ。　若林　拓（18）MZ-700/1500　茨城県

THE SOFTOUCH

●Hyperword

ちょっとハイパーなアイデアプロッセッサ

Ogikubo Kei
荻窪　圭

アイデアプロセッサ機能を持つ日本語ワードプロセッサ，Hyperword PRO-68Kが登場した。独自のウィンドウにユニークで多彩な機能を搭載。ハイパーな時代を予感させる新しいパーソナルビジネスツールだ。

X68000用　5"2HD版 2枚組　39,800円
シャープ　☎03(260)1161

2月号のCYBERNOTE PRO-68Kのレビューで，私は偶然にもサイバーの次はハイパーだと書いた（もちろん，その頃はHyperwordが開発中なんてこれっぽっちも知らなかった)。それがこんなに早く現れるとは思っていなかったので，いささか驚いている。果たして，本当にハイパワーなワープロが実現できたのだろうか。だとしたら，私は諸手をあげて大歓迎だ。それこそ私が望むワープロだったのだから。

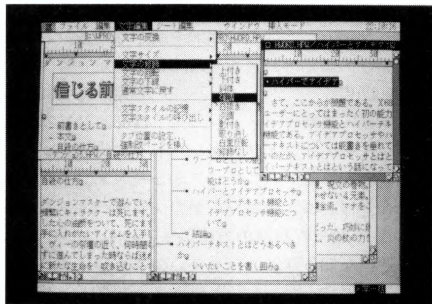
Hyperwordとは

Hyperword。それはマウスオペレーションを中心にした，ウィンドウばしばしのマルチウィンドウワープロである。

最初にいってしまおう。私の使ってみたところ，ウィンドウ自体のセンスは，なかなかよい。画面の具合も悪くない。X68000お馴染みの色変更もできるし，時刻も表示できるし，ウィンドウの背景の模様もいくつか変えられて悪くない。また，1行に占めるドット数を変えられる。20，24，28，32から選べる。1文字は16ドットなので，20ならば行間が4ドット，32ならば16ドットあくということだ。ビットマップディスプレイの勝利，というところだ。

複数の文書をいくつも開けるのもマルチウィンドウの強みである。異なった文書間でカット＆ペーストできるのも強みである。同時に開けるウィンドウは6つまでだが，それ以上開くときには自動的に古いウィンドウを閉じてから行ってくれるので，6つという数字は気にはならない。

ウィンドウにはズームアップ/ダウン，スクロールバーなどのほかに，アイデアプロセッサやハイパーテキスト機能で使用するアイコンを左下に持っている。なおウィンドウの最大幅は，文書の桁数に依存し，むやみと大きなウィンドウを開くことはできない。

操作の基本はプルダウンメニューである。右ボタンによるポップアップメニューはない。これには異論もあろうが，それはあとで書こう。また，ショートカットといって，ウィンドウを開かなくても，基本的な機能はキーボードによる入力でマウスに手を伸ばすことなく行える。プルダウンメニューもファンクションキーとカーソルキーを使えばキーボードから行える。

当たり前だが，FEPはASKである。本体付属ワープロと異なり，操作はHuman68k上でASKを使うときに準拠しているため，ユーザーコンフィギュレーションが効く。

ついでといっちゃなんだが，チャイルドプロセスを呼べる。これはうれしい。

ウィンドウが開く開く

文書の編集

ワープロの最低条件は，漢字かなまじり文が入力，編集できること，そしてそれを保存でき，印刷できることだ。さらに実用を考えるならば，操作性，親しみやすさ，速度，表現力が問題となる。そこから先はユーザーの用途次第だ。

Hyperwordの日本語入力はX68000標準のASKであるから，良くも悪くもASKだ。もちろん，インライン変換（カーソル位置での変換）は可能。辞書登録も登録キーで，名詞だけでなく，全品詞に対して行える。ただ，プルダウンメニューでの記号入力機能はつけてほしかった。

というわけで問題は文書の編集。編集機能は大雑把に分けて，**前へ行ったり後ろへ行ったり，切ったり貼ったり，捜したり置き換えたり**の3種類が基本である。さらに**線を引いたり揃えたり**がくる。これらは全ワープロソフトの必須課目で，倍角したり目次付けたりは付属機能だ。それぞれ見ていこう。

●前へ行ったり後ろへ行ったり

まずは“前へ行ったり後ろへ行ったり”，つまりカーソル移動や簡単な削除，挿入機能である。Hyperwordの場合，カーソルキーなど編集用特殊キーのほかは，プルダウンメニューにいくつかの機能が収められている。しかし，慣れてくればちょっとした作業でわざわざマウスに手を伸ばす面倒が実感されるだろう。そこで，**パワーキー**と称するエディタライクな操作がキーに割り付けられている。

とはいえ，コントロールファンクションはオリジナルな機能に割りふられているので，エディタに慣れたユーザーには誤操作が絶えないだろう。エディタライクな操作をしたいならば，XF1キーを使う。デフォルトでは，カーソル移動やバックスペース，削除，ロールアップ/ダウンのみが登録されている。ユーザー登録が可能なので，もっ

▶文法がなんだ！　古典がなんだ！　世界地図がなんだ！　方程式がなんだ！　重力がなんだ！　あーすっきりした。これが載るころには，もう決まってるだろうな。
峰松　剛志 (18) X1turbo III　茨城県

とエディタライクにできる。とても評価できよう。しかも，CTRLキーとXF1キーの役割を起動時オプションで逆にできるので，そうしておけばエディタ慣れしたユーザーでもカスタマイズで快適環境が得られる。

カーソルの移動規則もエディタライクで，改行マークより後ろにカーソルは動かないし，上下も改行マークより後ろを避けて移動する。ED.Xと違い，ちゃんと上下移動開始時のカーソル桁位置は覚えていて，移動途中にその桁より短い行があっていったん改行マーク上に移動しても，以後はもとの桁に復帰する。ワープロといえばフリーカーソルが常識だが，エディタライクなこの方式が好きな人は（私を含めて）結構いるのではないだろうか。

余談だが，ロールアップ/ダウンキースクロール方向が付属ワープロとは逆である。これはHyperwordのほうが一般的なのでしかたがないだろう。

●切ったり貼ったり

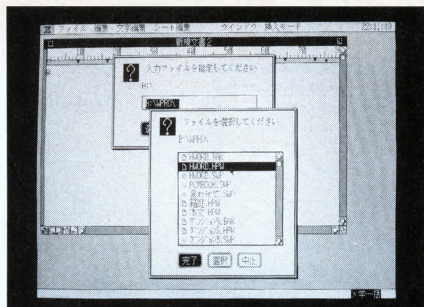
続いて，“切ったり貼ったり”である。マウスによるオペレーションが削除や移動，複写処理を“切ったり貼ったり”に変えてしまったので，このテーマは独立してしまった。付属ワープロでもこの快適さは際立っていた。で，右ボタンによるポップアップメニューのないHyperwordではどう処理しているか。

結論を先に言ってしまうと，処理の遅さやプルダウンメニューしかない点を補うための工夫はかなり見られる。ひとつは，形式的な範囲指定ならばマウスのドラッグを必要としないことだ。

たとえば，上の文でマウスのマの字の前で一度クリックする（カーソル位置の確定)。もう一度クリックすると，“マウス”という単語だけが反転する。もう一度クリックする。すると，“ひとつは”から“ことだ。”の1文全てが反転する。さらにクリックすると“　結論を”から“ことだ。”までの1段落が反転する。さらにクリックすると始めに戻るといったぐあいに自動的に範囲指定ができるのだ。もちろん普通にドラッグもできるため，使い分けると便利である。

指定した範囲はカットしたりコピーしたりできる。これもメニューバーまでマウスカーソルを動かしたくない人は，キーボードより操作できる。たとえばカットならCTRL＋Xというぐあいだ。範囲指定をしてCTRL＋Xすればそれは削除されてクリップボードに移動するのだ。コピーならCTRL＋C，ペーストはCTRL＋Vである。

ファイルの入力

印刷イメージのプレビュー

これをナイスアイデアと見るか苦肉の策と見るかは勝手だが，便利なのは確かである。また，範囲指定もキーボードで可能である。

もうひとつは，文の置き換えである。たとえば，上の文章の“ナイスアイデア”を“画期的”に直したいとき，付属ワープロでは削除と挿入を行う必要があった。しかし，Hyperwordでは“ナイスアイデア”を反転させ，そのまま“画期的”と入力すると，“ナイスアイデア”が削除されて代わりに“画期的”が入るのである。**感覚的に馴染みやすい**方法でとてもよろしい。

また，カット＆ペーストに使うクリップボードが4つ用意されているのもなかなかである。惜しむらくはクリップボードの中身を見る機能がないことだ。

●捜したり置き換えたり

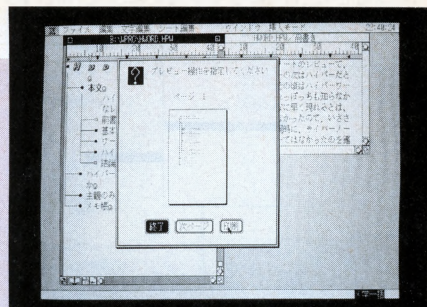
“捜したり置き換えたり”，つまり検索と置換はひと通り揃っているので，特に言及する必要もないだろう。検索/置換文字列はクリップボードから持ってくるか，キーボードから入力するか，前回のものをそのまま使うかのどれかである。これもキーボードでの実行も可能だ。

●割り付け

基本の説明はこんなところ。続いて，割り付けであるが，左揃え，右揃え，中央揃え，均等割り付けがある。これらは割り付けの情報を行に持たせるのではなく，いきなりスペースを適当に挿入して形を作ってしまうので，均等割り付けしたあと戻すことはできない。これは不便。

●罫線・装飾

線を引いたりするのは，付属ワープロのようなマウスオンリー方式ではなく，キーボードを使う。線種はX68000用だけあってさすがに多い。

字の大きさを変えたり修飾したりも種類が豊富にあり，白抜きや淡い色もある。X68000であるからどの修飾も画面上で確認できるが（PC-9801のようなテキストビットマップではないマシンになると，強調でさえ画面上では強調を表す記号をつけるにとどめているものが多い)，文字修飾した文字は表示速度が著しく低下してしまうのが問題である。なお，4種類まで任意の修飾の組み合わせたものを4通りまで記録できる文字スタイル記録機能もある。

保存と印刷

続いて，ファイル操作と印刷関係を見てみよう。

●ファイル

ファイルは“.HPW”という拡張子の専用の文書ファイルが作成される。これは付属ワープロの“.SWP”ファイルと異なり，テキストファイルではない。ただ，テキストファイルの読み込み，書き出しはできるので，他の文書を持ってくることも可能だ。せめて，SWPファイルをテキストファイルに変換して読む機能は欲しかった。付属ワープロでできることはほとんどすべてできるのだから。

なお，起動時オプションでバックアップファイル作成の選択ができる。

●印刷

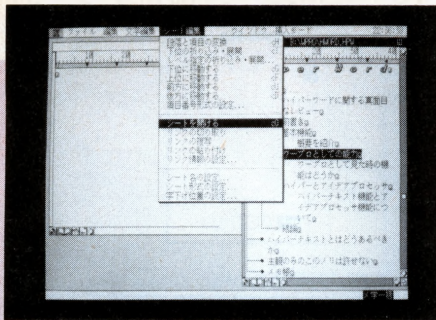
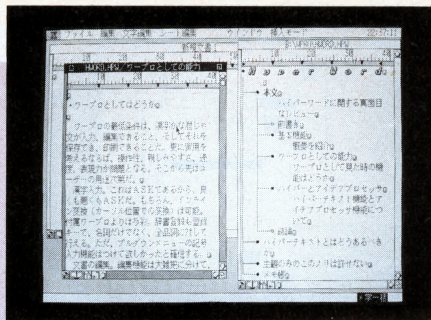
印刷機能である。文書の書式決定も印刷のメニューで行う。付属ワープロのように，マウスで簡単に文書の幅を変えることはできない。なんとかしてほしいところだ。

書式決定は結構面倒なので，標準書式というものを定められる。いつも使う書式は標準書式として登録してしまえば，つぎからは簡単に設定できる。あと，袋とじや2段組みができないのは残念。

さて，印刷である。印刷には文書の印刷とシートの印刷がある。

印刷には全ページ印刷と，1ページ飛ばしで印刷する機能がある。1ページ飛ばしで印刷するメリットは，両面印字したいときに発揮される。まず奇数ページだけ印字して，そのあと，裏に偶数ページだけ印刷するというアクロバチックな技が駆使できるのだ。

また，1ページごとにヘッダとフッタをつけることができるのであるが，そのヘッ

▶月刊マイコンではX1が「よー子ちゃん」と呼ばれています。なぜだろう？　僕は「えっちゃん」と呼ぶのに。
程田　勝也（16）X1F/turboZ　茨城県

目次から編集したいタイトルを選んでシートを開くと目的のウィンドウが表示される

ダやフッタの設定もかなり細かくまで可能である。付属ワープロでは文書名か章名かページかという数種類だけだったが，Hyperwordでは何行でも任意の文章を入れられる。さらに，いくつかの変数も用意されており，可変な情報をヘッダやフッタに入れたいとき（文書名や日付，ページなど）はそれに応じた記号をつけておくと，印刷時によきにはからってくれる。たとえば，印刷日付は{DATE}であり，ファイル名は{FILE}である。

なお，プレビュー機能として印字イメージを画面で確認できるのは（遅いけど）いいことである。

印刷といえば，使用できるプリンタであるが，Hyperwordではいくつかのページプリンタにまで対応しているのが新しい。X68000とページプリンタという組み合わせを使っている人はまずいないと思うが，今後少しずつ増えてくるだろう。

ワープロとしての評価

私はいまHyperwordを使って今月号の原稿を書いているわけだが，処理速度と現在行数（カーソル位置が何ページ目の何行か）がわからないこと以外に問題は感じられないどころか，なかなか快適である。後者はこういう仕事をしている人には必需品だが，総行数はHELPキーで教えてくれるし，一般のユーザーにとっては致命的な問題とはならないだろう。

Hyperwordは機能的には確実に付属ワープロを上回っている。問題は，付属ワープロの持つ身軽さ（機能が少ない代わりにプログラムサイズが小さく，処理が速い）や機動力をどう評価するかだろう。その点，Hyperwordの場合，多機能だけに速度的な難点もある。

処理速度については，エディタとしては使い物にならないくらい，DTPソフトと思えば結構速いな，というくらいである。付属ワープロが速すぎたのかもしれない。

プログラムが巨大なのでその都度ロードしなければならない機能は待たされてもある程度しかたがないし，Hyperwordはむしろそういった面ではキャッシュを多用しているので効率がよい。逆にワンウエイト入っているようなメニューの開き方や，漢字候補選択時の反応の遅さはなんとかしてもらいたいし，なんとかできそうなところである。なにより挿入やデリートの際の基本処理は遅すぎる。大量の文書を書く人には耐えがたいかもしれない。しかし，大半のユーザーはそうではない（と思う）ので，その辺はワープロ中心ユーザーが増えて不満が噴出するころまでに改善されればいい問題である。

X68000初のアイデアプロセッサ

さて，ここからが問題である。X68000ユーザーにとってはまったく初の能力，アイデアプロセッサ機能とハイパーテキスト機能である。アイデアプロセッサやハイパーテキストについては能書きを垂れてもいいのだが，試用レポートの本質から外れるので，ここでは具体的な機能を中心にまとめる。

世間一般でいうところのアイデアプロセッサ，あるいはアウトラインプロセッサには共通点がある。

まず，文章の階層構造である。文書はたいてい複数の項目に分かれ，それがさらに下位の項目を持ち，最下位に文章が入るという構造にすることができる。特に教科書や論文などはそういう構成だ。部や章があって項があるものだ。そういった文書を作成するとき，あらかじめ目次を作って全体の構成を考えておき，最後に各項目の文章を入れて完成という手法があると便利である(**トップダウン**)。これがひとつのアイデアプロセッサ機能である。

もうすこしくだけていうと，目次に当たるシートで思いつくままに項目を書いていき，あとでそれを並べ替えたり上位下位に分類したりして整理し，最後に各項目の文章を入れて文書を完成させる，という方法を支援する機能だ。**階層構造**というのは便利な考え方で，論文を書いたりマニュアルを作ったり企画書を書いたり考えをまとめたりするときに役立つ。独り仮想ブレーンストーミングとでもいったようなことができるのである。

この機能を使うと，ひとつの文書が複数のシートを持つことになる。**1ウィンドウ＝1シート**と考えてよい。最初に開いたシートは自動的に先頭シートとなるため，目次のような感覚で扱う。その下位に属するシートは任意の名前で任意の数だけ作ることができる。下位のシートだからといって他と区別されることはなく，長さの制限もないし，さらに下位のシートを持つことも可能だ。

原則として，下位シートを持ち得るのは段落と項目のみである。段落というのは改行から改行までだ。項目というのは，まあ，見出しと思えばいい。

ある段落を範囲指定して，シート編集から“シートを開ける”を選ぶ。すると，その段落に帰属するシートが開くのだ。そこへ文章を書き込んでいく。

上の写真で説明しよう。このレビューの目次なのだが，ツリー表示形式のシートで，段落ではなく項目にしてある。本文の下の“ハイパーとアイデアプロセッサ”というのがいまあなたが読んでいるこの文章である。本文と同じレベルでいくつかタイトルが書いてあるが，これはコラムになっている(はずだ)。ちなみに，項目の左にマークがあって，□は下位にシートがあることを，菱形は下位にシートを持っていないことを示している。つまり，まだ，見出しだけ決まっていて，書いてない文章というわけだ。

ここで，本文の左のマークにカーソルを合わせてダブルクリックすると，下位の見出しが見えなくなる。アイデアプロセッサにはたいていこういった下位構造を隠す機能があり，これは，項目がどんどん増えたときに全体が見通せなくなるのを防ぐためだ。

ハイパーテキスト機能とは？

さて，この文章が入っている“ハイパーとアイデアプロセッサ”を開いてみよう。それにはまず目的の項目で一度クリックする。すると，項目名が反転する。そこで，右ボタンを押しながら左クリックだ。すると，ぼよんとシートのウィンドウが現れる。普通はこのようにして使う。

つまり，**文書というものは1本の長い巻物（スクロール）ではなく，いくつものテ**

▶4月号は「なぜか」とても楽しかった。でも3月号のMIDI特集をまだまだやってほしい気もする。
毛塚　健次（17）X68000　栃木県

キストが集まってできたものだという考えを実践したものだ。さらにひとつの文書を構成するテキストはそれぞれがバラバラに独立したものではなく，常にどこかのテキスト（Hyperwordの場合はテキストのある段落か項目）と関係づけられている。

こうしてだんだんとアイデアプロセッサの話がハイパーテキストの話に移っていくわけだ。Hyperwordのハイパーテキスト機能とは，ハイパーテキストのような感覚で文書中の必要な部分を検索できる機能をさす。そのためにはアイデアプロセッサ機能の上下の階層構造だけでは弱いので，シート間リンクの機能が追加されている。任意のシート同士をつなぐことができるのだ。そうすると，シートCは先頭シートの下位シートであると同時にシートAの下位でもあるといったことが可能となる。

Hyperwordの，効率よく文書を書くためのしかけがアイデアプロセッサ機能であり，効率よく文書中の必要な情報を取り出せるしかけがハイパーテキスト機能である，といった認識が妥当だろう。

以上のワザのために，ウィンドウの左下のアイコンに上位，下位の移動や，シート形式を文書/ツリー，シート属性を本文/追記/参照/メモに分類できたりといった細かい機能も持っている。

ハイパーテキスト機能をうまく使えば，テキストデータベース的にも使えるというわけだ。

とりあえず総評である

私はこのHyperwordのベータバージョンを受け取って以来，全原稿をこれで書いている。多少速度に難があるにしても，いくつもの文書がマルチウィンドウで開けること，カスタマイズでコントロールファンクションによる編集ができること，アイデアプロセッサ的な考え方にすでに馴染んでいてとっつきやすかったことがあるだろう。結構気に入っていたりするのだ。

もっとも，私の環境は40Mバイトのハードディスクにプログラムを入れ，メインメモリは２Mバイト，その内256KバイトをRAMディスクにしてHyperwordの作業用ディスクとしている。そのくらいの環境は必要だろう。

当然注文もある。アンドゥ，リドゥ機能をつけてほしいということだ。特にアンドゥはいまや常識である。なお，マニュアルには書いてないが，うっかりシートを削除してしまっても，メモリ中には残っているので復活は可能だ。

では，どんな人にお勧めだろうか。まず卒論を書く人，あなたには非常にお勧めである。印刷時にはページ入りの目次までつけてくれる。

まとまった文章は書かなくとも，つい思ったことをメモにしたり雑文帳を作ったりしてしまう人，あなたにも有用だろう。そういったメモをまとめたり，短い文章を詰め込むにはHyperwordはうってつけだ。

DTPしたい人。ページプリンタをサポートしているとはいえ，レイアウト機能はさほどでもないので，DTP仕様のワープロが出るのを待たなければならないかもしれない。ただし，凝ったレイアウトをしたいのでなければ，字体の豊富さや罫線種類は並のワープロを軽く凌駕しているから満足できるだろう。

ビジネス文書を書く人。ハイパーテキスト機能をうまく使えば，たくさんある文書ファイルから使えそうな文書を探して，それを書き換えて，といった手間がかなり縮められる。

では，絶対にHyperwordに向かない人。それは，エディタで文書を書くのがいちばんだと信じて，実践している人だ。そういう人にはこの速度は耐えられないだろう。

というわけで，実をいうと細かい欠点は抱えているにしろ，新しい試みを持ったワープロということで私は評価している。これでバージョンアップを重ねて，世間の荒波にもまれて洗練されてくればかなりいいソフトに育つのではないだろうか。それが売れるソフトの第一条件だからね。

シート間のリンク指定

ハイパーテキストへの道

奇しくも先月号の117ページで，ハイパーテキストは文字だけを扱うものではない，と書いてしまったあとで，こうして文字だけを扱うハイパーテキストと銘打ったものが出てきてしまった。それに文句をつけてもいいのだが，大人げないので，とりあえず，“文字情報だけを扱うハイパーテキスト”と解釈しておこう。

では，ハイパーテキストとして見ると，Hyperwordはどうか。

まず，ハイパーテキストっていうのは，複数の互いに関係づけられたテキストの複合体だとする（正しい定義は別にして，まあこんなもんだろう）。この“関係づける”という妙な日本語がなかなかわかりにくくて難しい概念のような雰囲気をかもしだしているが，大雑把にいってしまうと，“線を引いてつなげる”ようなものだ。つまり，トップダウンにテキスト間のつながりを追うことによって。

こうなっているとなにがいいかというと，検索が便利であったり，常に必要な情報だけが見られるので，上手に作れば，無駄な部分を読まされずにすむのである。これが，１次元の文章を２次元に拡大するということだ。

で，ハイパーテキストの一般的なイメージは，文章の中である単語をクリックするとそれに応じた説明なり下位の文章なりが現れる，といったものだ。しかし，Hyperwordでは段落，あるいは項目にしか下位のシートを持たせられない。ここがいちばん気になるところだ。

たとえば，下位構造を持っている単語は四角で囲むなりしてマークしておき，それをクリックすると次のシートが現れる，という動作は，ハイパーテキストと称するいろんなソフトがやっている。できればHyperwordにもそこまでやってほしかったものだ。

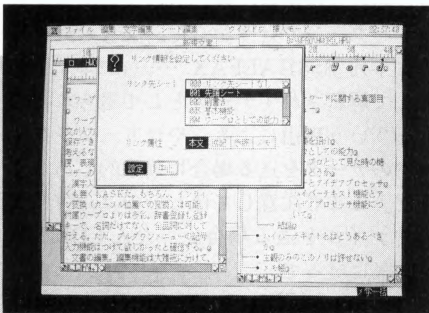
さらに，関係づけが上位/下位だけではなく，もっといろんな関係（並列/類似/反対/参考など）ができれば，画期的だ。フレキシブルなデータベースともいえるだろう。

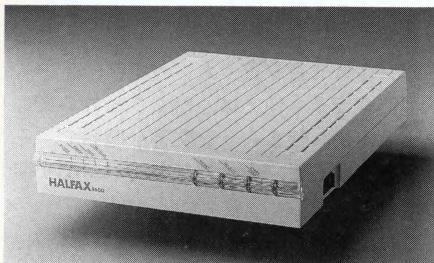
そのうえで，ハイパーテキスト作成支援機能ももっと充実させてほしい。たとえば，文書中のシート間の関係が全部わかるシートマップ機能とか（それで，見たいシートをダブルクリックすればそれが開くとか），そのマップでマウスで線を引けば関係づけができてしまうとかだ。ここまでいけば，なにもいうまい。ほとんどエキスパートシステム開発支援ツールのノリだけど，エキスパートシステムやフレーム理論，意味ネットワークの考え方は結構ハイパーなので，ワープロに応用できたらかなり面白いのではないだろうか。

で，ハイパーテキストだが，せっかくX68000はテキストビットマップなのだから，モノクロの画像情報をテキストに持たせるのは困難ではないと思う。さらに，OPMのテキストならCRTではなくOPMドライバに出力し，PCMならAD PCMに出力し，とやれば，もう立派なハイパーテキストではないか。X68000だったらできるはずである（せめてCPUはMacintoshポータブルと同じCMOS版68000-16MHzじゃないときついか）。そうすると，使う人にとってみればテキストと同じ感覚でいろんな情報を見られるようになるのである。

そうなると，Macintoshのハイパーカードに対抗して，X68000はハイパーテキストだ，などといえたりするだろう。いまいったようなことを，情報をテキストでなくカードのように持ったものがハイパーカードだといえるからだ。

アイデアプロッセッサというと変わり種のワープロといったイメージがあるかもしれないが，Hyperwordの持つ方向性にはこれからも大いに期待したいところである。

▶大学に合格したっ。これもすべて皆さまのオカゲです。で，皆さまってだれでしょうね？
中村　正和（15）X1turbo III　群馬県

新　仲夫

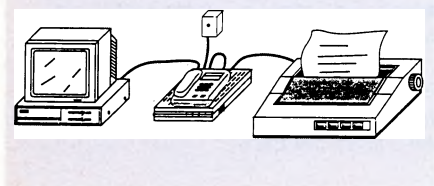
パソコンFAXアダプタ
HALFAX9600/9600EX

3月号のペンギン情報コーナーでも紹介したが，HAL研究所からFAXアダプタが発売された。発売されたのはバッファメモリ容量256Kバイトモデルの「HALFAX9600」（78,000円）と768Kバイトモデルの「HALFAX9600EX」(98,000円)の2製品。X68000やPC-9801などで使用できる。

なにが目新しい?

FAXアダプタとは，パソコンのデータをそのまま相手先のFAXに送信できる装置だ。通常ワープロの文書を送る場合「プリンタ出力→FAX送信」という手順が必要だが，FAXアダプタを使った場合プリンタ出力の手間が不要だ。さらにFAX送信の「スキャナ読み込み→画像変換→送信」のうちスキャナ読み込みも不要であるのでFAXに比べ鮮明な画像が送れる。

当然のことであるが，FAXアダプタはFAXをエミュレートするだけである。よって，FAXアダプタを使ってパソコンのデジタルデータを送信した場合でも，受信データは送信データとは異なる「FAX形式のデータ」である。これは送受信の両方がFAXアダプタであっても同じことだ。

現在，特定マシン専用の拡張ボード型FAXアダプタは多く発売されている[1]。HALFAXはボード型FAXアダプタとは異なり多数のマシンで使用できるという特徴を持つ。具体的には「セントロニクスまたはRS-232Cインタフェイスを備え，PC-PR系またはNM系のプリンタに出力できるパソコンやEWS」であればいい。つまり送信はプリンタに印刷する感覚で行えるわけだ。

図1　HALFAXの基本的な接続図

また受信も可能で（パソコンを使用中でも受信可)，受信データは直接プリンタに出力できる。通信速度はG3標準の4800bpsに加えて，ビジネス用G3の9600bpsにも対応(相手先の速度に応じて自動的に切り替わる)。図1にHALFAXの標準的な接続方法を示すので参照していただきたい。

HALFAXはパソコンから印刷命令があれば，印刷データをFAX形式データに変換したうえで送信する。印刷データがグラフィックデータの場合は単純にFAX形式に変換するのだが，テキストデータの場合はHALFAXが持っている24ドットフォントのイメージでFAX形式に変換する。そこで，グラフィックデータを送る場合は元イメージに忠実であることが予想される。

接続するにあたって……

接続は「基本的には」非常に簡単だ。パソコンのプリンタケーブル（プリンタ側）と電話線(電話機側)をHALFAX本体に差し込むだけである。適当な穴があるからすぐわかるだろう。ただ，接続にあたってはほかに以下のことを注意する必要がある。

1）　モデムがつながる電話回線であること

4線式（4つの線が着信，発信，内線，電源に割り当てられているもの）ではなく2線式の電話回線でなければいけない。これは難しく考えることはない。使えないのはオフィスなどのビジネスホン，極端にいってしまえば内線が使える電話機のみだと考えて間違いないだろう。よって会社などで使う場合は外部直通の電話回線を使うこと。一般家庭用の電話回線は基本的に2線式なので，問題なく使えるだろう。

2）　X68000ではアースをつける

X68000に限らずシャープの製品はアースがついているものが多いが，なぜかHALFAXはプリンタケーブルのアース線をX68000本体のアース端子に接続しないと動かない。これはマニュアルにも載ってないので覚えていたほうがいいだろう。

3）　プリンタドライバを組み込む

Human68kではPC-PR用プリンタドライバはPRNDRV2.SYS，NM系はPRNDRV3.SYSなのでいずれかを組み込めばいい。なお，Z'sSTAFF PRO-68Kなど自分でドライバを持っているソフトを使う場合も組み込むこと。HALFAXの送信コマンドをテキストで送る必要があるからだ。

4）　本体のディップスイッチを設定する

2番(ダイヤル/プッシュ回線)，8番(PC-PR系/NM系）などの切り替えは注意したほうがいいだろう。

使用レポート

では，実際に使ってみよう。今回は以下のテストを行った。

1）　テキストファイルの送信
2）　WP.Xの文書の送信
3）　Z'sSTAFFの画像の送信

●テキストの送信

図2のようなファイルを送った。最終行の「$$$HALFAX SEND 02394159」は送信コマンド（数字は0発信の電話番号）である。この文字列が見つかった時点で文書の終わりと見なされ送信が開始される。逆にいえばこの文字列が送られるまで文書はHALFAXのバッファにたまっていくのみである。

HALFAXを制御するコマンドは，$$$HALFAXのあとに続くテキスト文字である。このコマンドは結構豊富で，親展（4桁のパスワード)，時間設定，電話番号登録などの機能もある。では，プリンタで印刷したあと通常のFAXで送ったもの（図2）とHALFAXで送ったもの（図3）とを見比べてほしい。HALFAXで送ったもののほうが鮮明ではあるのだが，HALFAXのフォントを展開しているので直接プリンタに出力するものとは若干イメージが異なっているのがわかる。

●WP.Xの文書の送信

次に図2のファイルにWP.Xで作った外字を入れた文書を送ってみた。図4にプリンタで印刷したあとFAXで送ったものを，図5にHALFAXで送ったものを示す。この結果，元の文書のイメージは損なわれず，かつ鮮明であることがわかる。元の文書に忠実な理由は，WP.Xの文書出力がドットパターンで行うからだ。文書の最終行の文字列「$$$HALFAX ……」がコマンドとして解釈されずデータとして送られたのもそういう理由による。やはり，グラフィックのデータを送る場合に効果的であるようだ。ここで注意してほしいのは，WP.Xのファイルを送る場合いったんCOMMAND.Xに戻って，

ECHO $$$HALFAX SEND 02394159 > PRN

のように「文字列」をプリンタに出力しなければいけないことだ。

●Z'sSTAFFの画像データ送信

グラフィックに威力を発揮するということを圧倒的に感じるのがこれだ（図6プリンタで印刷した画像，図7さらにそれをFAXで送ったもの，図8 HALFAXで送ったもの)。画像の劣化がほとんどないどころか，直接プリンタに印刷するより鮮明だ。「電話線を通さずFAXに送ればページプリンタとしても使える」(U氏)といった意見が出るのももっともだ。直接プリンタに印刷するより鮮明な理由は，スキャナ読み込みがないことに加えてFAXのプリンタの解像度のほうがパソコンのよりも若干優れているからだ[2]。ここでもいったんZ'sSTAFFを終了したうえでコマンドラインからSENDコマンドを送る必要がある。

あと，HALFAXの送信時の注意として，

1) バッファ容量以上のデータは送らない

2) A4サイズ以上の文書を送らない

などがある。しかしモノクロということもあり256Kバイトモデルで送れない画像データなどはあまりないだろう。また，ワープロなどから直接出力するときの書式設定も注意すること。

これからの周辺機器

HALFAXは，市販のFAXアダプタと同程度の価格で，マシン汎用性があることから，コストパフォーマンスは高いといえる。ただ，バッファ512Kバイトで2万円違うというのは高いような気はするが。

あと，欲をいえば，

1) モデム機能

2) 受信データのパソコン取り込み機能

などがあればもっと喜ばしい。両方とも少し改良すれば実現しそうだが，どうだろうか？　この2点が実現すれば非常に汎用的な通信機器が誕生すると思うのだが。しかし，こういったマシン非依存でなおかつ高機能な周辺機器は通信機器に限らずもっと増えてほしいものである。

＜問い合わせ先＞

㈱HAL研究所　☎03(252)5561

1)　X68000用ボード型FAXアダプタとしては，シャープから「CZ-6BC1」(79,800円）が発売されている。

2)　通常市販FAXの場合，スキャナ読み込み精度，プリンタ印刷精度とも200dpi（8ドット/mm）程度。これに対してパソコンのシリアルプリンタは160〜180dpi（スキャナは300〜400dpi程度）となっている。

図2　プリンタ出力後FAXで送信

```
<漢字と英数字のみのファイル>
(出力 COPY DOC PRN)
 1234567890!"#$%&'()=
 ABCDEFGHIJKLMNOPQRST
 abcdefghijklmnopqrst
 ＡＢＣＤＥａｂｃｄｅ
 あいうえおかきくけこ
 亜胃鵜絵御歌木句毛個
<文書の終わり>

$$$HALFAX SEND 02394159
```

図3　HALFAXで送ったテキスト

```
<漢字と英数字のみのファイル>
(出力 COPY DOC PRN)
 1234567890!"#$%&'()=
 ABCDEFGHIJKLMNOPQRST
 abcdefghijklmnopqrst
 ＡＢＣＤＥａｂｃｄｅ
 あいうえおかきくけこ
 亜胃鵜絵御歌木句毛個
<文書の終わり>
```

図4　WP.Xの出力をFAXで送信

```
<外字、漢字、英数字の文書>
(出力はWP.Xで行う)
 1234567890!"#$%&'()=
 ABCDEFGHIJKLMNOPQRST
 abcdefghijklmnopqrst
 ＡＢＣＤＥａｂｃｄｅ
 あいうえおかきくけこ
 亜胃鵜絵御歌木句毛個
<以下、外字>
 Oh!X68000
 xyz
<文書の終わり>
$$$HALFAX SEND 02394159
```

図5　WP.Xから直接HALFAXに送信

```
<外字、漢字、英数字の文書>
(出力はWP.Xで行う)
 1234567890!"#$%&'()=
 ABCDEFGHIJKLMNOPQRST
 abcdefghijklmnopqrst
 ＡＢＣＤＥａｂｃｄｅ
 あいうえおかきくけこ
 亜胃鵜絵御歌木句毛個
<以下、外字>
 Oh!X68000
 xyz
<文書の終わり>
$$$HALFAX SEND 02394159
```

図6　Z'sSTAFFの出力結果

図7　図6をFAXで送ったもの

図8　Z'sSTAFFから直接HALFAXに送信

▶私，28才，一児のパパです。最近1歳8ヵ月の息子にX68000を奪われている。彼は，スーパーハングオンがずいぶん気に入っているらしく，彼が寝るまでX68000に触ることができません。
高島　和典（28）X68000　群馬県

ラジコンスティックの製作

Kuwano Masahiko
桒野 雅彦

サイバースティックにコンパチのアナログジョイパッドまで現れて，アナログスティックはもはや選ぶ時代です。今回はラジコン用のプロポを使ってワイヤレス化に挑戦してみましょう。同時にアナログスティックの謎にも迫ります。

常に動き，変化する物。変化を止めたミニチュアたち。ガラスケースの中の止められた時間の流れにあって，積もった埃の崩れる音だけが現在を伝える。

凍結された瞬間（とき）から無限の空間を描き出す模型と動き続けるなくてはならないラジコンの対比。

甲高いエンジン音を響かせて地を走り，水を切り，空を駆けぬけるラジコン模型。コントローラ一式で何万円もすることや，広場がだんだん減ってきて遊べる場所がなくなってきたことから動いているのを目にする機会は少なくなったとはいうものの，街の模型屋さんのショーケースでは最上の場所に置かれ，まわりのプラモデルなどとは役者が違うと寡黙に主張するラジコン機器。

止められた時計が再び動き出す。人の夢の中でしか動けなかった模型達に生命（いのち）が宿り，人の意志のままに動く。そのとき、人は傍観者であると同時にレーサーであり，船乗りであり，パイロットとなる。

ああアナログスティック

パソコンの世界では，昨年のサイバースティックによってようやくまともなかたちのアナログジョイスティックが登場したわけですが，模型の世界ではマイクロプロセッサが世の中に登場する以前からすでにスティックの傾きをアナログ的に渡すプロポーショナルコントロール（比例制御）が実用化されていました。

これがラジコンスティックの全貌だ

このことからすると，有線式のアナログジョイスティックが日の目を見たのが1989年とは，ずいぶんと待たされたものです。

それだけにサイバースティックがパソコン界に与えた衝撃はかなり大きかったようです。やはりスイッチだけのジョイスティックではもの足りないと，ゲームを作っている側でも感じていたのでしょう。いまではPC-9801用のゲームでも“アナログジョイスティックが必要です”というものが出始めています（説明書ではしっかり“サイバースティック”と書いてあるところが笑える）。

サイバースティックのもうひとつの功績は，これまでのジョイスティックポートになんら特別な細工をすることなく，4チャンネルのアナログ情報と14ビットのデジタル（スイッチ）情報を渡せる方法を提唱したことです（仕様書ではデジタルスイッチは10ビットですが，実際には14ビット分送っています）。現在発売されているサイバースティックでコントロールできるのはアナログ3チャンネルとデジタル10ビットですから，まだまだ拡張の余地があることになります。

ジョイスティックポートということを忘れて，純粋に4チャンネルのアナログデータと14ビットのデジタルデータの受け渡しと考えてみると，ずいぶんと広い応用が見えてくる気がしませんか？

今回は，この受け渡しの実験を兼ねてアナログスティックの先輩であるラジコンのプロポをX68000に接続してみることにしました。もちろん，接続といっても，プロポを改造するわけではありません。プロポの出す電波を受け，それをサイバースティックと同じタイミングで渡してやろうというものです。名づけてラジコンスティック。

使ったプロポが2チャンネルのものですのでスロットルの操作はできませんが，スイッチのほうはサイバースティックにあるものはすべて使えるようにし，ついでにフットスイッチもつけてみました。いつでもフルスロットルになるのを我慢すればアフターバーナーもちゃんと遊べますヨ。

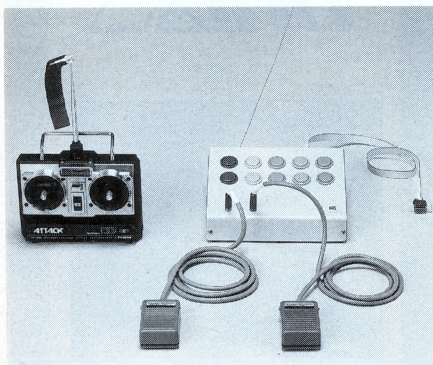
プロポの選択

プロポと接続するといっても，いちばん肝心な情報であるプロポがどのようにしてスティックの傾きを伝えているのかといった事柄については公開されていないようです。本を見つけようにも，そもそもラジコンの本が対象にしているのはラジコン模型なのであって，プロポではないのですから，探すだけ無駄な気がします。

だいたい，これだけ一般的になっているワイヤレスリモコンにしてあの状況だったことを考えると，まず絶望的と考えるのが妥当でしょう。と，いうことになれば……えーい，しかたがない。やっぱり自分で調べるしかないじゃないか。ということで，例によってシンクロのお世話になることとなりました。

自分で調べるとなれば，まずはプロポを手にいれないことには話になりません。ふらりと秋葉原まで遊びにいったついでに模型屋を回りながら機種と値段をチェックしていたら，とある店で送信機単体（受信機とサーボモーターはなし）で1,000円という，超特価品が目に止まりました。メーカーはこの業界ではよく知られた双葉電子工業です。シリーズ中では最下位機種にあたる2チャンネルのラジコンカー用のプロポです。その場で買い込んでしまいました。

形は写真を見てもらえばよいでしょう。型名はFP-T2NBLというもので27MHzのAM，と本体と箱に書いてあります。買いにいくときは型名はもちろんですが，27MHzというのを必ず確認するようにしてくだ

▶ゲームさえも多角的に特集してくれるOh！Xがボクは好きだ。
中島　潤史（15）X68000PRO，MZ-2000　埼玉県

さい。昔はラジコンといえば27MHz帯ばかりだったのですが，この周波数帯域はCBや漁業無線などと同じ周波数帯域であることなどもあって妨害や混信が多いため最近は40MHz帯もよく使われるようです。

気になる入手性と値段についても調べておきました。入手性のほうは，特に問題はなさそうでした。各店のショウケースを見た感じでも双葉電子工業の製品がいちばん品ぞろえがよいようでしたし，最下位機種というのはだいたいどこでも置いてあるようですから手に入れるのは容易でしょう。最近になって自宅の近所の模型屋なども覗いてみましたが，ラジコンを扱っているところならば必ず置いてあるようでした。

もうひとつの懸案である正規購入(？)時の値段ですが，だいたい1万円以下にはなるようです。ちなみに私が買ったその店では（もちろん，受信機，サーボモーターつきのものが）6,800円でした。

プロポはどうやって動いているの

買ってきたFP-T2NBLは箱を見る限り27MHzでAM方式であるということです。AMというのは音の大小にあわせて電波の強さを変えてやる方法のことで，振幅変調ともいわれます。普通の中波や短波のラジオはこの方法で音声を送っています。ちなみにFMというのは周波数変調のことで，こちらは電波の強さは一定のまま，音の大小にあわせて周波数を変えるというものです。振幅が一定ですから，雑音の影響を受けにくいというメリットはありますが，受信機の回路はAMよりもやや複雑になります。

AMというなら電波の強さが変わるのだろうから，シンクロで眺めてもわかるかな？と思って試してみました。結果は上々。ちゃんと波形を見ることができました。

振幅変調というよりは，27MHzのON/OFFといったほうが近い感じです。スティックを動かしながら波形を見ていたら，データの送り方はわりと単純であることがわかりました(図1参照)。

プロポからの電波は基本的に一定の強さで送信しっぱなしの状態になっていて，この中に一定周期で3回，電波が途切れるところが作られています。この途切れの間隔がスティックの傾きに対応しているのです。右のスティック（左右方向）を動かすと，最初の途切れたところから，2番目のところまでの幅が変化します。2番目と3番目のあいだは左のスティック（上下方向）の傾きに応じて変化します。

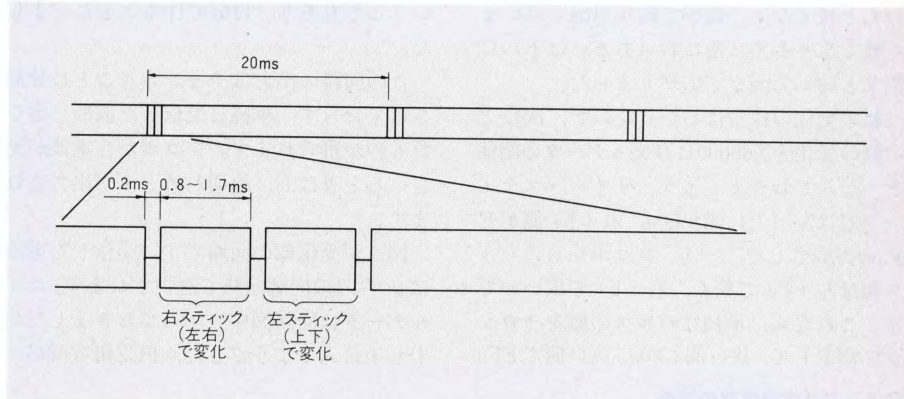
ラジコンの受信側では，このパルスの幅を取り出してそれぞれのチャンネルのサーボモーターに与えるようになっています。サーボモーターは自分の回転角と，与えられたパルスの幅の比較を行って誤差を修正する方向に回転し，一致すると停止するようになっています。ラジコン模型では，これを使って舵を切ったり，速度の調整をするようにしているわけです。

つまり，このプロポを使ったラジコンでは，次のような流れで情報が伝えられていることになります。

スティックの傾きの変化
｜
途切れと途切れの間隔の変化
｜
（電波）
｜
〈受信機〉
サーボモーターに与えるパルス幅の変化
｜
サーボモーターの回転

FP-T2NBL

今回のラジコンスティックでは，この最終段であるサーボモーターへのパルスの幅を数値に直し，それをサイバースティックの仕様にあわせてX68000に渡せるようにすればよいということになります。

図1　プロポ(FP-T2NBL)の送信波形

ラジコンとデータ転送

電波を使い，離れたところからものを自在にコントロールできるラジコン技術の進歩はそのままデータ伝送の進歩でもありました。

いまでこそ，電波を使ってヘリコプターでさえも自在に遠隔操縦できるラジコンも，最初の頃は単なるトーン信号を使ったON/OFF制御しかできませんでした。電波で「ピー」という音を送り，受ける側では「ピー」というのが聞こえたら（受信できたら）リレーをカチッと動かすというだけですから，遠隔操縦というよりも遠隔スイッチといったほうが正しいようなものでした。

そのうち，回路技術やICの普及によってコントローラのスティックの傾き具合に応じて動くことができるようになってきました。これが比例制御（プロポーショナルコントロール）です。いまや，ラジコンのコントローラを差す言葉になっている「プロポ」の用語はここからきています。

さらに時代が進んでくると，ICの集積度も上がり，そしてついにマイクロプロセッサの登場を迎えることになります。マイクロプロセッサを使い，デジタル通信でプロポからのデータを送るようにすれば，スティックの傾きにとどまらず，さまざまな情報を送ることが簡単にできるようになります。伝送の誤りのチェックや修正もお手のものですから，多少ノイズの多い場所でもバタバタしたりせず，安定して動かすことができます。これが，ラジコン界でPCM方式と呼ばれているものです。

PCM方式をデジタル式，それまでのプロポーショナル制御のものをアナログ式と読み換えれば，一般的にデジタル-アナログの比較として行われていることがそのまま当てはまります。デジタル側は外来ノイズなどに対して強く，いろいろな情報を含めることができるというメリットはあるものの，電気的に複雑になる（高価になる）ことや，数値化する都合上，どうしてもある程度カクカクとした動きが出てくるなどのデメリットがあります（なくそうとするとまた高くなる）。

いまのところ，デジタル(PCM)方式が使われているのはプロポのなかでも高級機の部類だけで，個人がホビーとして気楽に購入できる価格帯のものはほとんどアナログ方式のものになっています。

▶吉田幸一さん，お帰りなさいっ！　K君シリーズの復活を期待しています。
今野　和浩 (19) MZ-2500　埼玉県

波形の計測

次にプロポの送信波形をもう少し詳しく見てみましょう。まず，スティックから手を離した状態で波形をとってみます。観測は前回と同様，シンクロのプローブの先に簡単なループを作って，プロポのアンテナに近づけて行いました。

送っている周期は約20msでした。1秒に50回ほど送っているわけです。電波が途切れる時間は0.2ms，2つある途切れと途切れの間隔（パルス幅）はどちらもだいたい1.3msといったところでした。

次に，スティックを傾けて測定してみます。先ほども触れたとおり，右のスティックを動かすと最初のパルスの幅が，左のスティックでは2番目のパルスの幅が変化しますが，送ってくる周期や電波が途切れる時間にはほとんど影響がないようです。

パルス幅は，左（あるいは上）に倒していくと狭くなり，最小で約0.8msくらいまで狭くなります。逆に右（あるいは下）に倒すと約1.7msまで広がりました。

幅の変化の様子はわかったので，次にこの幅の変化とX68000に与えるデータの関係をつかんでおきましょう。サイバースティックでは左（上）側が00H，右（下）側がFFHの方向でした。一方，プロポからのパルス幅は左（上）で狭く，右（下）で広いのです。これなら，単純にパルスの幅をカウンタで測定して，狭い側で00H，広い側でFFHになるようにするだけでよさそうです。

気になる中間点も，1.3msというのは0.8と1.7の平均値程度であることから，特に小細工はしなくても問題なく中央にきてくれそうです。

偶然の一致なのか，それとも設計した人がラジコンの事情を知っていたのか，単にVRAMのXY座標にあわせただけなのかわかりませんが，うまくしたものです。動きがわかったところで，いよいよ回路の設計にとりかかることにしましょう。

プロポ受信回路の設計

まともに買ってきたプロポには受信機がついています。この出力はサーボモーターとつなぐため，各チャンネルごとに分割されています。この出力をカウンタで計測し，それぞれのチャンネルのデータとする手もあるのですが，自作品の中にブラックボックスがあるというのはあまり面白くないということもあり，自分で作ることにしました。

この回路の出力はチャンネルごとに分割されておらず，単純に受信した波形に応じたものが得られます。プロポから電波が出ているときは0，途切れると1が出力されます。

図2が受信部の回路です。受信した電波はまずT1の同調回路で選択されます。コイルデータは回路図中に示しておきましたが，もし手に入るようでしたらFCZ研究所のハムバンド用コイル（28MHz用）を使っても構いません。

選択された電波は2SK241と次段の2SC945で増幅されます。FETである2SK241はゼロバイアスでもいいのですが，普通のバイポーラトランジスタである2SC945は軽くバイアスをかけておく必要がありますので，100KΩの抵抗でバイアスをかけ，数mAのコレクタ電流を流しておきます。

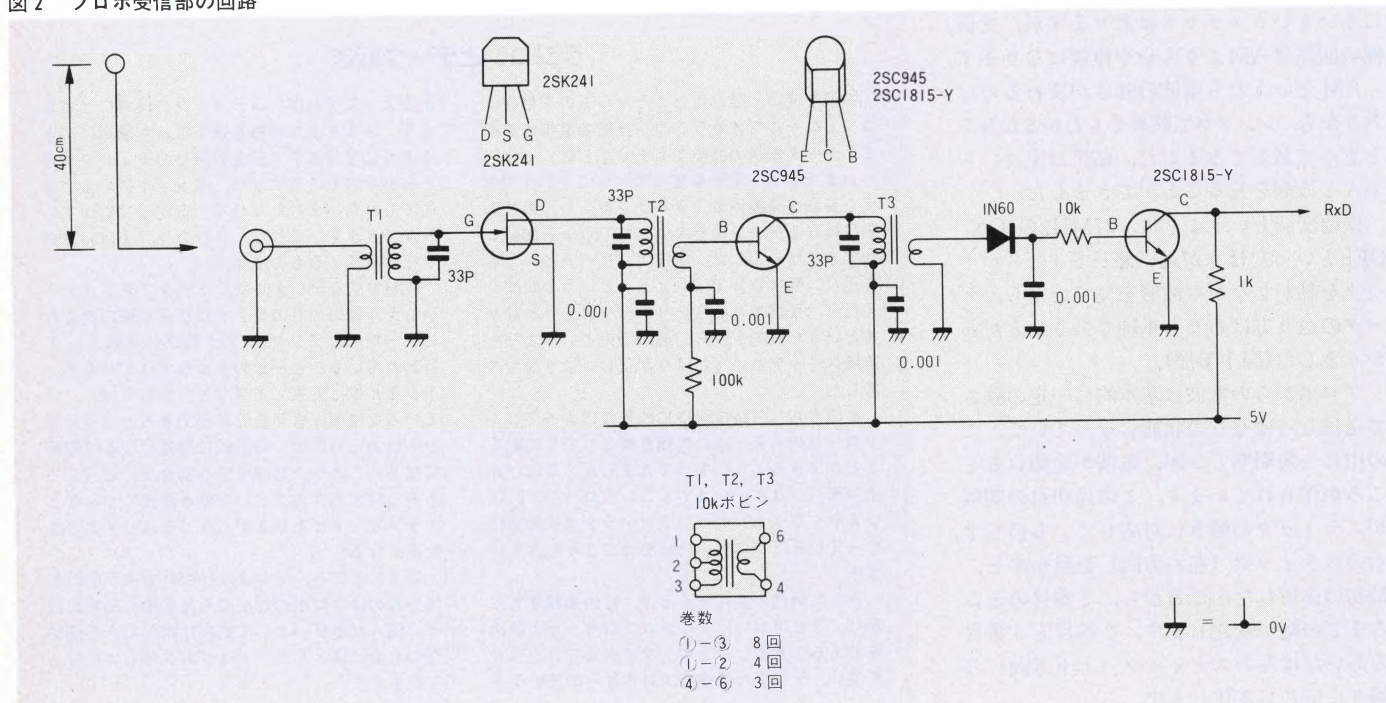
十分増幅された電波は1N60で検波され，この出力を使って次段の2SC1815をスイッチングします。プロポからの電波がない状態ではトランジスタはOFFしているので出力は1（Highレベル）になっています。電波がくるとベース電流が流れるのでトランジスタがONとなり，出力が0になります。

もう少しまめにやるなら，軽くAGCをかけて検波出力を安定させるべきなのでしょうが，実際のラジコンのようにX68000が走り回るわけでもありませんし，私たちも動き回りながら操作するわけではないので，簡単に増幅するだけですませています。

受信波形処理回路の設計

受信回路から出力される波形はプロポからの電波がきていると0，途切れると1になるようになっています。それでは，パルス幅を数値化する回路を考えてみましょう。基本はパルスの幅すなわち電波が途切れ，再び出始めたところから，次に途切れるまでの時間を計ればよいわけです。ただ，単

図2　プロポ受信部の回路

▶このたび，私の友人O君が晴れてX68000ユーザーになりました。これまで自慢してきたかいがありました。
菅沼　光剛(16) X68000　埼玉県

純にカウントしたのでは幅が最小のときに0にならなくなりますから，あらかじめ幅が最小のときの分を差し引いてから測定することになります。図3は，この部分の考え方を示したものです。

途切れを検出したら，最小値分だけ遅らせてカウンタをスタートさせます。次の途切れがきたらその時点のカウンタの値をラッチします。少し遅らせてカウンタをリセットし，次のチャンネルのカウントに備えます。

このままですと，どこかで途切れのカウントを間違うと以後はずっとずれたままになってしまいますので，どこかでつじつまをあわせる工夫をしておかなくてはなりません。今回は途切れが一定時間以上こなくなったことを鍵として使うことにしてみました。

図1を見てわかるとおり，プロポの発信している波形は1回のデータ転送を行ったあと次のデータを送ってくるまでずいぶん長い待ち時間があります。データを表すために使っているパルスの幅が2ms以下なのに対してデータ転送の間隔は 20ms もあります。ですから，たとえば途切れが見つかってから 10ms もたつのに次の途切れがこなければ回路をリセットするように作れば，たとえどこかでつじつまがあわなくなったとしても，次の転送は正しく受け取れるようになるわけです。

それでは実際の回路図を見ていきましょう。図4が受信波形処理回路，図5，6がその動作タイミング図です。この回路では受信回路からきた信号RxDを元にして，カウンタのリセット信号（CCLR1）とカウント値のラッチ信号（CL01～CL31）を発生しています。

2チャンネルなのにラッチ信号が4つあるのは，将来同じシリーズの4チャンネルプロポ（T4NBL）が手に入ったときを想定したもので，実際に使うのはCL01とCL11の2つだけです。

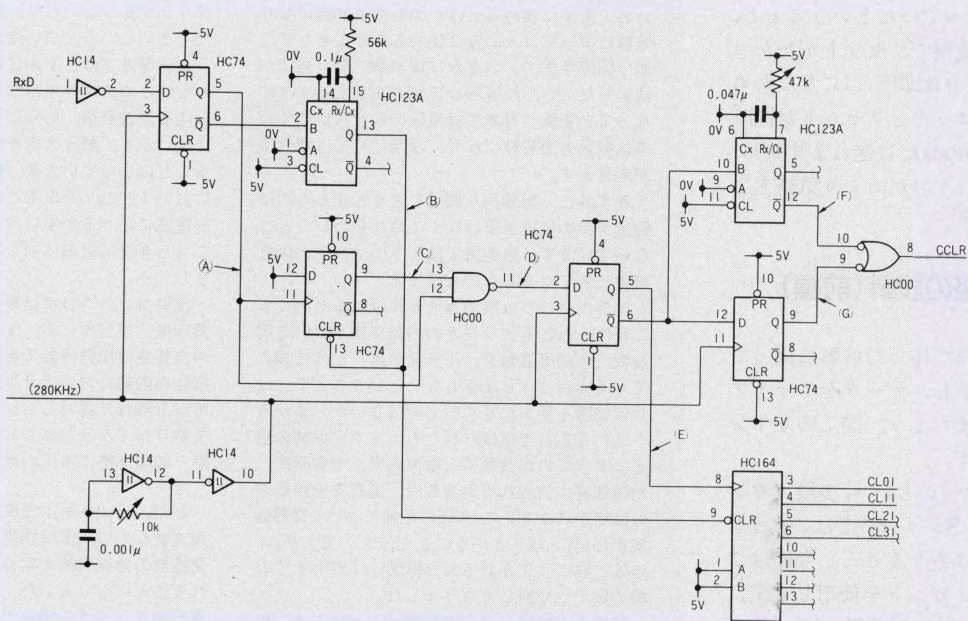
まず，左下のHC14が2つ並んでいるのが発振回路で，ここで作られた一定周期のクロックがラジコンスティック全体の動作タイミングを決めています。この周波数はプロポのパルス幅を使って調整します。プロポの送ってくるパルスの幅は最小時で約0.8ms，最大時で約1.7msでした。

幅の変化は1.7－0.8＝0.9msです。これを256等分するクロックは0.9/256＝0.00352msの周期，つまり約280KHz となります。HC14 を使った図のような発振回路では発振周波数はだいたい 1/CR になります。ここでコンデンサを0.001μFとすれば，抵抗値はR＝1/(280K×0.001μ)ですから，約3.6KΩになります。パルスの幅はプロポの具合などによって変化しますので，それにあわせてクロック周波数を調整できるように10KΩの半固定抵抗を使いました。

受信回路からきた波形はいったん，HC14で波形整形されたあと，HC74によって内部のクロックに同期させ，扱いやすくしてお

図3　受信波形からデータへの変換の考え方

図4　受信波形処理回路

▶おおっ，荻窪氏も「MASTER キートン」読んでいたのかぁ……。あの漫画は「ちびまる子ちゃん」，「マリオネット師」とともにイチ押しです。もしかして，勝鹿北星とはズバリ，キートン先生そのものでは？
木藤　将也（14）X1turboZ　埼玉県

きます。

その右側のHC123はプロポからのパルスの途切れが一定時間続いているかを見るもので，約5msの間途切れがこなければ回路をリセットします。

パルスがきているあいだはカウンタを動かし，値をラッチさせる信号を作ります。HC123の下のHC74とその隣のHC00でちょっとした細工をしてスティックの情報を持つパルス幅の部分を切り出した波形を作り，さらにHC74で受けて1クロック遅らせます。

この右上にあるHC123は受信されたパルス幅から最小値のときの分をキャンセルするものです。この幅も調整できるようにしてあります。HC00の出力が0になってからこのCRで決められる時間，HC123の出力は0になっているため，CCLR1が1になり，カウンタはリセットされたままになります。HC123の出力が1に戻るとカウンタが計数を開始するわけです。

HC00の出力の立ち上がりがカウントの終了を示すことになります。順序として，まずカウンタにラッチ信号を送るため，HC164にこの波形を与えます。さらに次のチャンネルのカウントに備えるため，カウンタのリセット信号をラッチ信号から1クロック遅らせて送ります（HC164の上にあるHC74）。

HC00とHC164は直結してもよさそうに見えます（たぶん直結しても大丈夫でしょう）。ここでわざわざHC74で遅らせたのは，HC74が1ゲート余っていたということと，HC123による時間監視でリセットがかかったときにHC00の出力は即座に1になるためにHC164にはクロックとリセットが同時（と，いってもHC00の分だけ遅れますが）に入ることになるというのがあまり気持ちよくなかったからです。

カウンタ回路の設計(前編)

受信波形処理回路で作った波形に従ってパルス幅をカウントし，データをラッチする回路を見ておきましょう。図7の左半分がこれにあたります。

HC4040は12ステージという，多段カウンタです。普通の74LSシリーズではこれほど大きい段数のものはありません。今回はこのカウンタの下位8ビットを使用します。このカウンタの値が次のHC374でラッチされます。HC4040の下にあるHC30は，カウンタの値がFFHになったら，それ以上カウンタが進まないようにするためのものです。

図5　受信波形処理回路の動作(1)

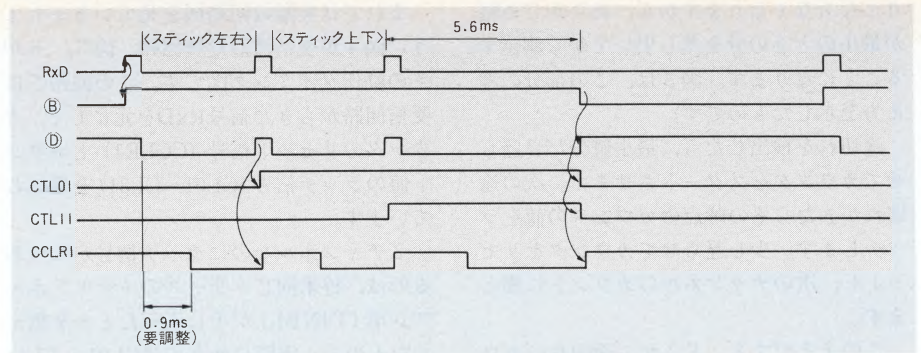
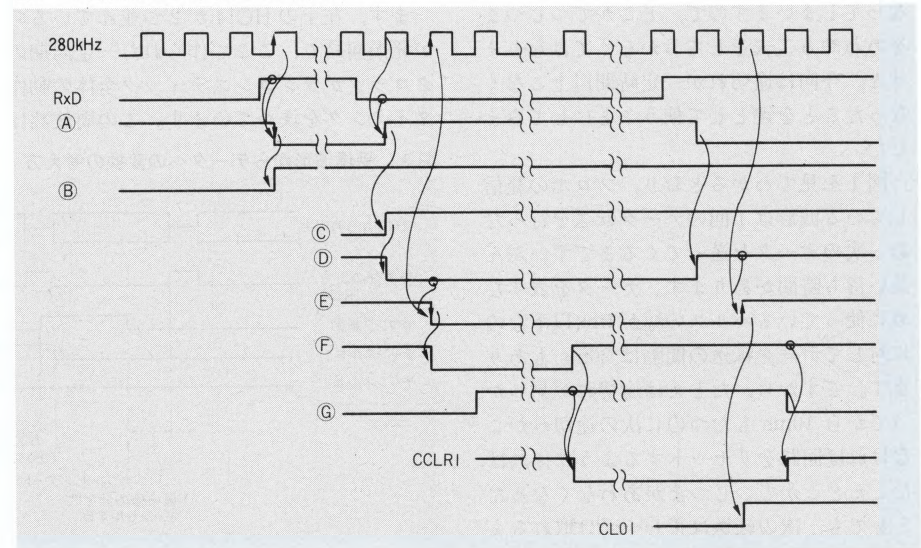



図6　受信波形処理回路の動作(2)

周波数の割り当て

電波というのは全世界で使われるものであるため，勝手に決めるとほかの交信の妨害になり，場合によっては人の命に関わることにもなるため，国際条約でおおまかに周波数の割り当てが決められ，さらに各国が詳細を決定するようになっています。日本では電波の取り扱いについては郵政省の管轄になり，法規としては電波法があります。

基本的に，無線局を開設しようとするものは郵政大臣の免許を受けなくてはならないことになっています。無免許で使えるのは，この例外規定にあてはまるものだけです。

例外のひとつは微弱電波と呼ばれるものです。これは，もともとラジオの内部で使っている周波数（中間周波数や，局部発振器）が外に漏れてくるのはどうしようもないということで，ほかに妨害を与えたりする心配はないからよかろうということで設けられたものです。以前の規定はすべての周波数帯（電波天文，宇宙研究，地球探査に使われる周波数や，遭難安全用周波数は除かれます）で100mの距離における電界強度が15μV/m以下ということだったのですが，4年ほど前に改正されて322MHzから10GHzまでの周波数がやや厳しくなりました。

部屋の中で使うなら微弱電波で十分です。以前は積極的な利用法としてはワイヤレスマイクくらいしかなかったのですが，近頃ではコードレステレフォンやヘッドフォンステレオなどの家電製品に組み込まれたり，パソコン関係でもプリンタをつないだり，RS-232Cを飛ばしてみたりといろいろ面白い使われ方をしています。

微弱電波ではさすがに弱すぎるということで，もうひとつの例外規定があります。こちらはあらゆる周波数帯，あらゆる電波形式というわけにはいかず，郵政大臣が用途と電波形式を定めることになっています。電界強度は500mの距離において200μV/m以下ということですから，微弱電波に比べるとずいぶん強い電波になります。こちらを微弱電波に対して小電力と呼んでおきましょう。

ラジコンのプロポは電波法施行規則で「模型飛行機，模型ボート，その他これに類するものの無線操縦用発振器であって，壁に囲まれた建築物の内部において又は建築物から500メートル以上離れた場所において使用するもの」として割り当てが行われており，13MHz帯，27MHz帯，40MHz帯に専用の周波数が割り当てられました。

ただ，40MHz帯は医療用の機器などと共用の周波数なので，混信妨害を防ぐためのラジコン発振器の推奨規格を定め，適合証明試験が行われることになりました。同時にラジコン模型の運用者をラジコン操縦士（なんか，カッコいい）として登録し，掌握することになりました。この目的で作られたのが財団法人日本ラジコン模型安全協会です。

▶「ストライダー飛竜」と「究極タイガー」を，X68000用に移植してくれないかなー。
寺門　正樹（17）X68000　埼玉県

カウンタはクロックの立ち下がりで進むタイプなので，ここがFE$_H$からFF$_H$になったとき，クロックは0(Lowレベル)になっています。そこで，FF$_H$になったらHC4040のクロック入力を強制的に0に固定してしまうわけです。これでカウンタにはクロックが入らなくなり，カウントはFF$_H$で停止します。カウンタにクリア信号が入るとカウント値が00$_H$になり，HC30の出力も1になり，再びクロックが入ってくるようになります。

ホストインタフェイス回路の設計

カウンタ回路の右半分にいく前にホスト，つまりX68000とのインタフェイス部分の回路設計を行ってしまいましょう。

このブロックはホストからのREQ信号を受けて動作を開始し，サイバースティックに似せたタイミングでホストにカウンタやスイッチの値を渡すところです。サイバースティックではこの処理をワンチップマイコンで行っているようですが，それほど大したタイミングではなさそうなのと，自作品でマイコンを使うとROM(CPU？)ライタを作ったり，プログラムを作るといった手間がかかることから今回はランダムロジックで処理することにしました。

サイバースティックでは転送速度も何通りか選べたりするのですが，遅いモードというのはCPUの能力が低かったり，不定期にDMA転送が割り込まれたりするために最速モードではついてこられないコンピュータ向けに作られたものです。X68000なら最速モードで十分ついていけますから，今回は最速モードだけを対象にすることにしました。もし遅いホストマシンにつなぐ場合は，このブロックに入っているクロック(280KHz)だけを半分なり1/3なりにしてやれば転送タイミングを遅くすることができます。

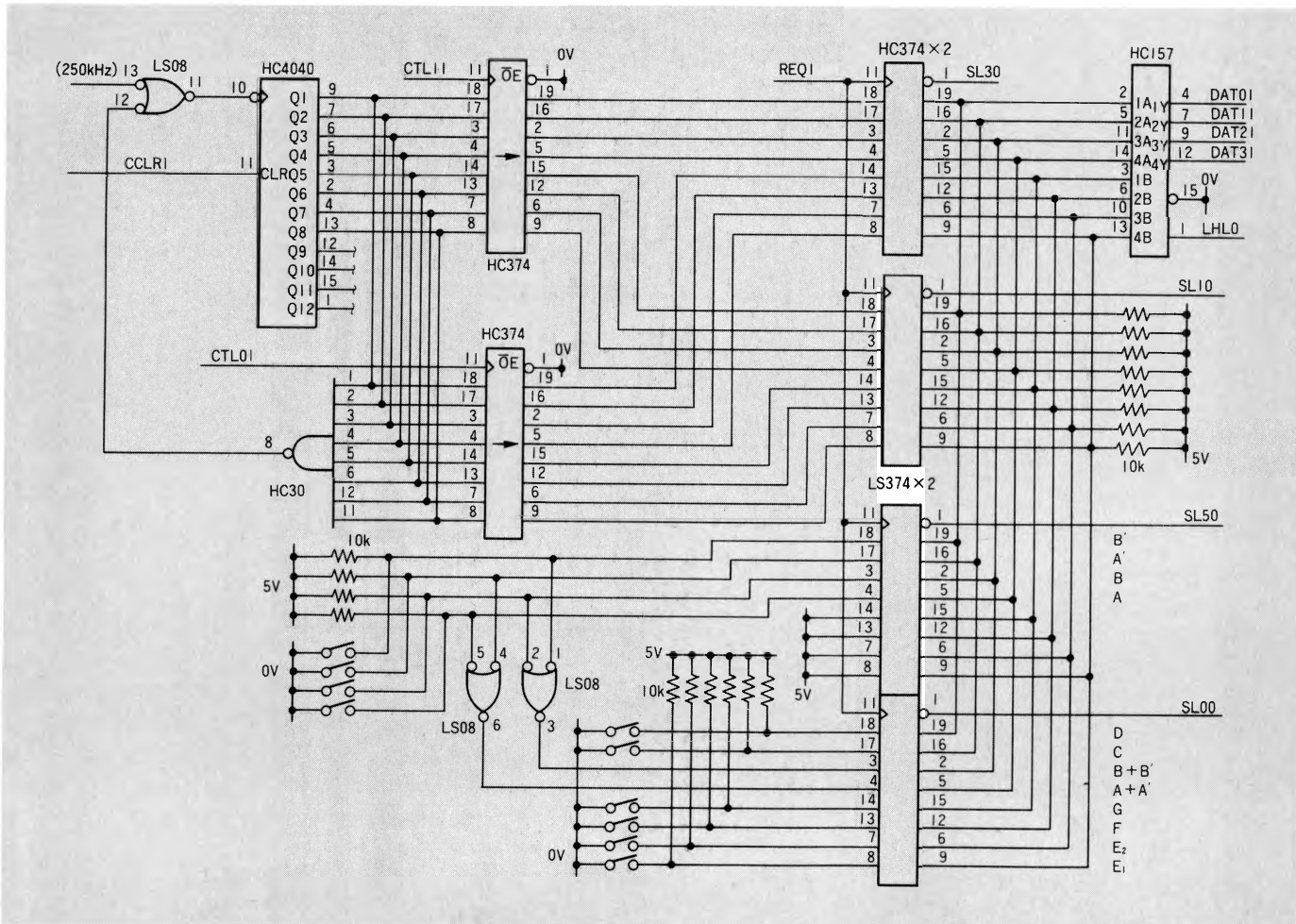
図8がこの部分の回路，図9，10が動作タイミング図になっています。図9はデータ転送全体を見たところ，図10は1回（1

HC123のタイミングに注意

74HC123にはどうやら出力パルス幅の係数によって2種類あるようなのです。これから自作に励まれる方は注意してください。

74LS123では，出力パルス幅は0.45×C×Rで計算される値になります（TI社のデータブックによる)。また，74HCシリーズも東芝の1985年版のデータブックでは0.46×C×Rとなっています。当初，この回路を実験していたときには手持ちの都合で74LS123で組んでいたのですが，作りなおしたときにはHC123にしました……すると，波形がまるで違うのです。パルス幅が当初の設計の2倍以上に伸びています。一瞬，コンデンサを疑ったのですが，2回路ともというのが納得できません。まさかと思ってパルス幅を測定して，係数を求めるとどちらも1.0。つまりパルス幅が単純なCとRの積になっているのです。試しにソケットを使って回路を組み，買ってきたHC123をいくつか差し換えたのですが，どれも係数は1.0で動いています。

実験したICのメーカーはカタログと同じ東芝です。カタログが間違っているのかなぁ？　とじっくりカタログと現物を比べると，現物にはTC74HC123の後ろに“A”がついているのに気がつきました。どうやら“A”がついたときに係数が1.0に変更されたようです。74HC14を使ったタイミング作成でも単純なCとRの積になりますから，たぶん係数が1.0のほうが作りやすいのでしょう。

図7　カウンタ周りの回路

組）の転送のタイミングを拡大したものです。図11は転送しているデータのフォーマットです。

ホストと接続される信号はREQ0，XACK0，XHL0，そしてカウンタ回路のDAT01～DAT31の計７本です。

動作タイミングの作成には，波形処理回路で作った約280KHz のクロックをそのまま使っています。このクロックはスティックからの受信データによって調整することになっていますから，それに応じてホストインタフェイス回路のタイミングも変化してしまうことになります。うるさい人でしたら顔をしかめるかもしれません。確かにあまり気持ちのよいことではないのですが，サイバースティックのドライバのソースリスト（昨年７月号）を見てもわかるように，ホスト側のタイミング管理はかなりいい加減で，要は規定時間以内（これもまた結構長い時間なのですが）に送り終わればそれでよいということになっています。

まぁ，そうはいってもあまり極端に違うと気持ちが悪いのでなるべく近いタイミングになるように工夫してみました。図９にクロックを280KHz としたときのタイミングとサイバースティックの仕様書上の値を書いておきましたので参考にしてください。

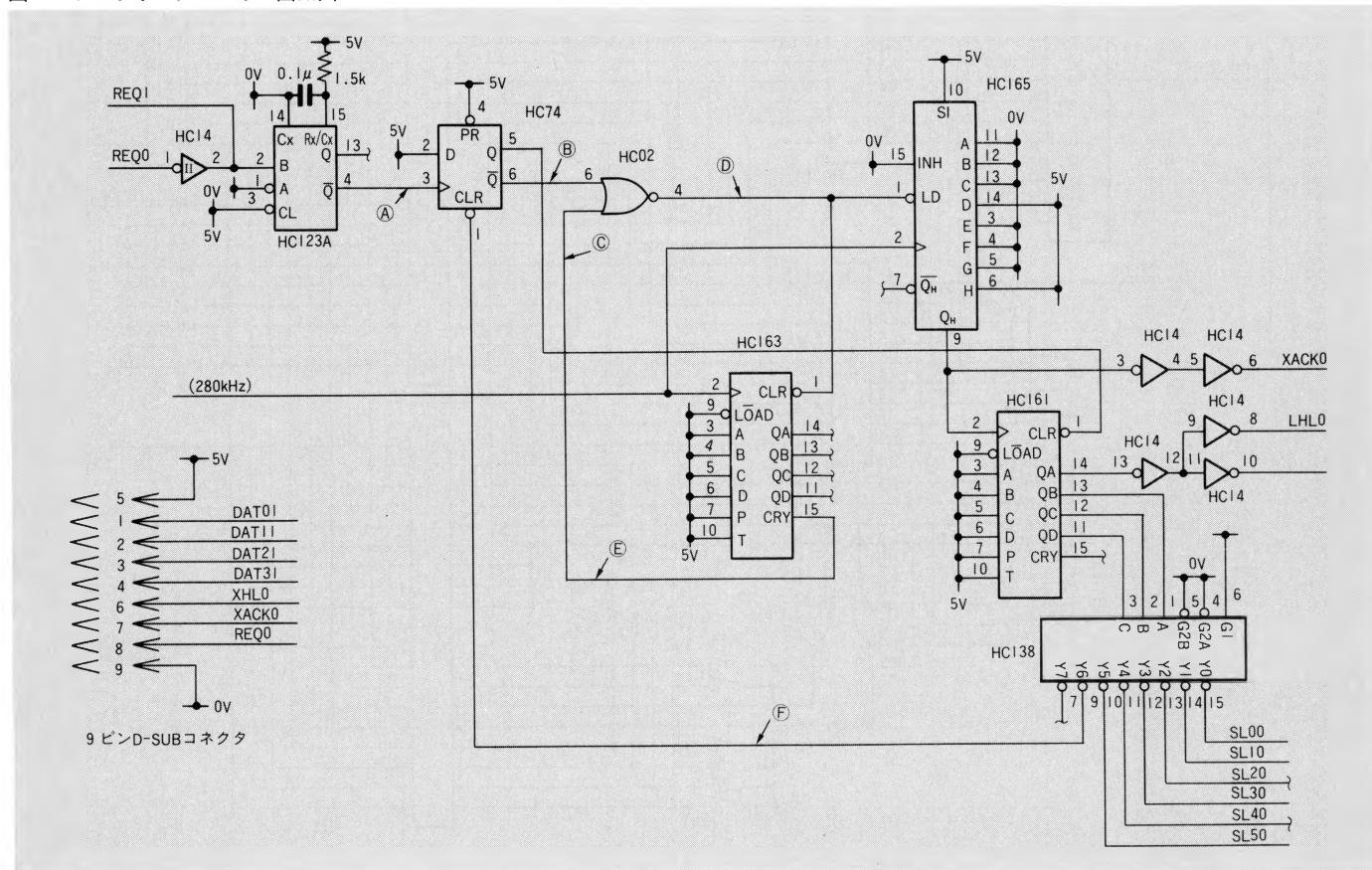
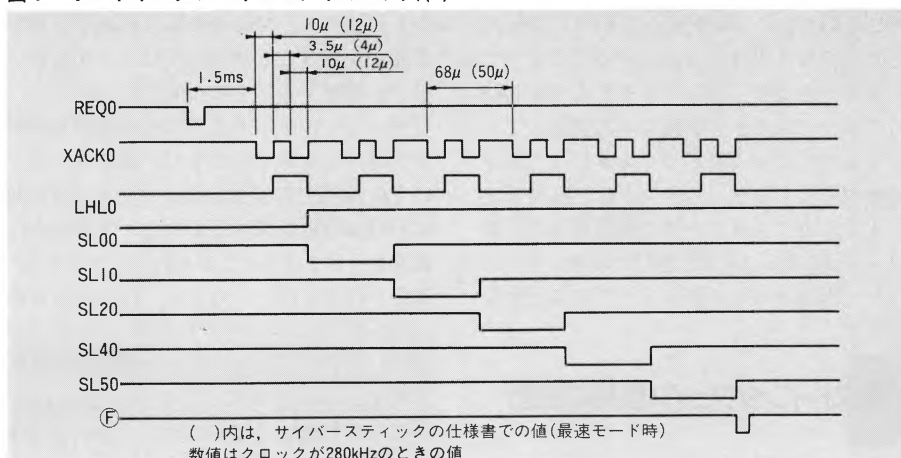
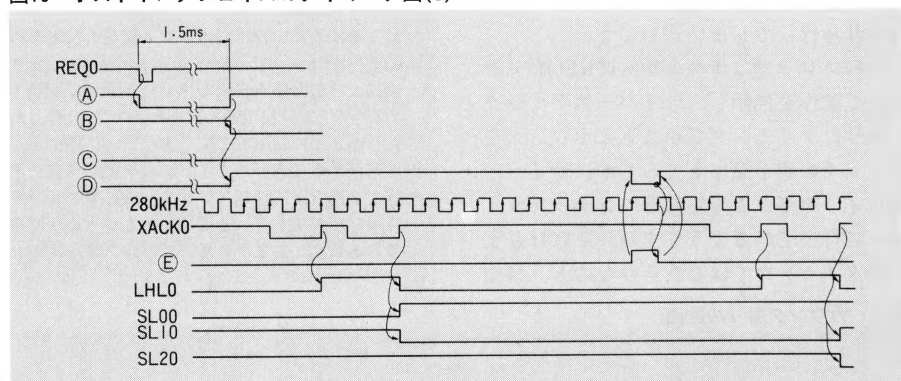



図９　ホストインタフェイスのタイミング図(1)

図10　ホストインタフェイスのタイミング図(2)

図８　ホストインタフェイス回路図

▶３月号のハミダシ「レンズをガイガー管に近づけると……」は，私のズイコーでも鳴る頻度が多いような気がする。　増原　しげお（24）X68000PRO-HD　千葉県

データ転送間隔の68μというのが，仕様の50μと比べてやや違いが大きいと感じられるかもしれません。対策がないこともないのですが，このままでも特に問題なく動いているのでそのままにしてあります。

それではホストインタフェイス回路の動作を見ていきましょう。設計はわりと苦労したホストインタフェイス部なのですが，できた回路はそれほど複雑ではありません。左端のREQ0というのが，X68000からの転送開始要求で，これが0になった時点からタイミングがスタートします。いきなりHC123のワンショットでタイミングを遅らせています。

当初，このワンショットはつけていなかったのですが，アフターバーナーを動かしながらサイバースティックのタイミングを調べているとき，REQを0にしてから1.5ms程度遅れてからデータの転送が始まっていることがわかったからです。おそらく，サイバースティックのCPUがREQを検出してからデータ転送ルーチンに移るまでの時間遅れなのでしょう。CPUを積まないラジコンスティックでは放っておくと即座に動作を開始してしまうので，ここで時間稼ぎをしているわけです。

次のHC74は，データ転送中を示す信号を作るものです。REQがくると，HC123の分だけ遅れてこれがセットされ，全データの転送が終わるとリセットされます。図面中央のHC163は2回のデータ転送ごとに少し時間をとるものです。サイバースティックでは，データの転送は2回を1組として送るようなタイミングになっています。この2回分のデータ転送のACK（X68000への応答信号）を作っているのが右上のHC165です。

ACKの回数はその下のHC161でカウントされ，最下位が偶数番目か，奇数番目かを示すビットに，そして2ビット目以降で何組目の転送なのかを示すのに使われます。これがHC138で分解されるわけです。12回の転送が終わるとY6が0になり，左上のHC74がリセットされ，回路全体が初期化され，転送終了となります。

カウンタ回路の設計（後編）

さて，先ほどやり残していたカウンタ回路（図7）の右半分を見ておきましょう。左半分のラッチ（HC374）にはプロポが送ってきたスティックの傾きのデータが入っています。このデータをホストからのREQ信号がきたときに右のラッチに取り込みます。こうしておかないと，ホストにデータを送っているうちに，データが変化する恐れがあるからです。

配線がクロスしているのは，ホストへの転送順序がまず各チャンネルの下位ニブル（4ビット単位のデータ）をすべて送ったあとで上位ニブルを送るようになっているのに対して，この回路ではひとつのラッチの下位ニブル，上位ニブルという順序で送っているためです。

スイッチのほうもデータをラッチさせています。SL50という信号がつながっているLS374の入力4本が5Vにつないでありますが，これが幻の12回目のデータになっています。ここにほかの入力と同じようにスイッチなどをつけてホストのソフトで対応してやればちゃんと使えます。

この回路ブロックではラッチアップ対策を手抜きするためにスイッチ周りだけLSシリーズを使いました。作るときに間違えないように気をつけてください。

製作

ガイガーカウンタのときのような特殊な部品はないので，部品集めで悩むことはあまりないでしょう。ICよりもむしろスイッチのほうが難しいかもしれません。私は，秋葉原のヒロセパーツセンターの入り口のところでザラザラっと並べられているジョイスティック用のスイッチを買ってきて並べ，さらに外部スイッチ端子をつけてフットスイッチをつけられるようにしてみました。

その他，ケースや電源ランプ，アンテナ端子などは各自の趣味で選んでください。スイッチもゲーム用なら2つで十分かもしれません。

製作は受信回路から片づけてしまいましょう。この部分は仮にも高周波を扱いますから，グランド（0V）のパターンをなるべ

図11　サイバースティックのデータフォーマット

回	XHL	DAT31	DAT21	DAT11	DAT01
1	0	A＋A′	B＋B′	C	D
2	1	E_1	E_2	F	G
3	0	チャンネル♯1上位			
4	1	チャンネル♯2上位			
5	0	チャンネル♯3上位			
6	1	チャンネル♯4上位			
7	0	チャンネル♯1下位			
8	1	チャンネル♯2下位			
9	0	チャンネル♯3下位			
10	1	チャンネル♯4下位			
11	0	A	B	A′	B′
12	1	〔幻の4ビット〕			

スイッチは押すと0，離すと1になる
A＋A′(B＋B′)はA(B)とA′(B′)のいずれかが押されていれば0になる
サイバースティックではFはSTART，GはSELECTボタンになっている
アナログデータは各チャンネルとも8ビットあり，4ビットずつに分割して送られる

サイバースティックのデータ転送

サイバースティックのデータ転送のやり方については昨年7月号や電脳倶楽部でも公開されていますが，若干間違いというのか，書き足りない点があるようなので復習を兼ねて触れておきましょう。

ホストが出力してくるのはREQ0のみで，サイバースティック側ではこれをスタート信号としてほかの6本を使ってホストにデータを送るようになっています。まっとうな伝送なら，互いに相手が受け取ったか否かをチェックしながらやるのですが，サイバースティックではREQを受けつけたらあとはスティック側はデータを一方的に垂れ流し，ホストはそれをせっせと読み取るという方法をとっています。

LHL0は今回のデータが偶数番目のデータなのか，奇数番目のデータなのかを示すもので，XACKはDAT01～DAT31にデータが設定されている（したがって，ホストはデータを読まなくてはならない）ことを示す信号です。

送ってくるデータのフォーマットは図11のようになっています。サイバースティックの仕様書では1回目のA＋A′，B＋B′のところが単にA，Bとなっており，また転送は11回で終わりになっています。F，Gは仕様書上はこの呼び方ですが，サイバースティックではSTARTとSELECTスイッチにしているようです。

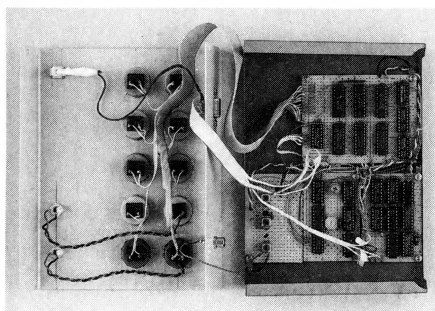
A＋A′，B＋B′はソフトの仕様ではA＋A′，B＋B′になっています。アフターバーナーのinput testでもちゃんと区別されていて，A，Bがセレクト用に，A＋A′，B＋B′がトリガON/OFFとして機能しています。

転送回数が11回というのは間違いのようです。当初，私もこのハード仕様書を信じて11回で転送をやめるようにしていたのですが，自作の読み込みプログラム（昨年9月号）ではちゃんと動くのに，アフターバーナーはまったく動いてくれず，ずいぶん悩まされました。シンクロを使ってサイバースティックのタイミングをもう一度全部チェックしていってやっと気がついたという次第です。そのつもりでドライバのリストを読んでみると，しっかり12回読み出しているのです。12回こないとエラーとして処理されてしまっています。たぶん，アフターバーナーもこれと同じようにしているために私の回路では動いてくれなかったのでしょう。11回で止めるのをやめ，12回送るようにした（実はこのほうが簡単にできる）ら無事に動くようになりました。

最後の12回目になにを渡しているのかは不明です。ドライバのリスト上でも読み出してはいるものの，完全に無視しているデータになっています。将来の拡張用かもしれません。

▶僕はいつもプレゼントより，ここに書く文がOh！Xに載っているかのほうが気になって，気になって……。
林田　和也（17）X68000，X1C　千葉県

基板のようす

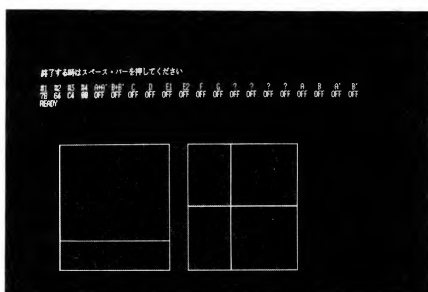
動作チェック

く広くとるようにします。私は昨年作った乱数発生器で使ったのと同じサンハヤトの504EGを使い，例によってカッターでパターンを切りとって必要なパターンを作りました。部品配置などはまぁ，信号の流れどおりにすればよいでしょう。

残るデジタル部の製作は，まず信号の流れを考えながらICの配置を決めていきます。ICの5VとGND(0V)のあいだにはIC1個，ないし2個にひとつずつくらい，0.1μFくらいのセラミックコンデンサをつけておきます。74HCシリーズに限らずC-MOSのICは動作時と非動作時の電流の変化が74 LSシリーズなどのTTLに比べて大きいため，電源ノイズが乗りやすいので，コンデンサはまめにつけてください。このようなコンデンサをパスコン（バイパスコンデンサの略）と呼びます。

各ICの空きピンのうち，入力になっているものは5Vか0Vに接続して入力がふらふらしないようにしておいてください。

また，私の作ったものでは各回路ブロックごとに基板を分けていますが，これは回路を製作/調整をするときに各ブロックごとにやっていったときの名残りです。これから作られる方は1枚の基板にまとめたほうが，基板間のつなぎの線がはい回ることもなくてよいでしょう。

調整

できあがったら，まず電源ラインをテスターでチェックします。ショートはないか，電源の配線を間違えていないか，念入りにチェックしておきましょう。実験用の電源を持っていればそれで5Vを，なければ十字を切ってX68000のジョイスティック端子につないで（過大電流が流れるとX68000の電源自体が出力をカットするので，本体が壊れることはないでしょうが）電源スイッチON！　まともなら，まず受信回路のコイルの調整からいきましょう。コイルの調整には金属のドライバは使えません。プラスチックでできたコアドライバと呼ばれるものを持っていればいちばんよいのですが，なければマッチ棒の先を削ったものなどでもなんとかなります。

アンテナ端子に10cmくらいの電線をつけておきます。IN60のカソード側（コンデンサのある側）の電圧をテスターで測りながら，プロポの電源を入れます。テスターの針が少し振れるはずです。この状態でコイルの中のコアを回していって，振れが最大になるところを探します。コアが入ったところと出たところの2カ所でピークがありますが，コアが入ったほうを使うようにします。3つのコアを最良の位置にすると，最初の振れ方からは信じられないくらいの電圧になっているはずです。

ここで，プロポのスイッチを切ります。もし，針が戻らないようなら，受信回路が発振しています。コアを最高点からずらして発振しないようにしてください。受信している電波と同じ周波数で2回も増幅しているのでやや発振しやすいのは事実です。

さて，問題のデジタル部の調整はX68000に登場願うしかありません。リスト1のアセンブラで書いた外部関数とリスト2のBASICプログラムを入力してください。基板上の半固定抵抗はとりあえず真ん中あたりにしておきます。プログラムを走らせると，読み取ったデータが表示されます。「入力エラー」の表示が出るようなら，ホストインタフェイス回路周りをチェックしてください。なんらかの入力データが表示されたら，まずはスイッチを試してみます。

押したスイッチが正しく読めたら，次にカウンタの調整にはいります。プロポのスティックを動かしてデータが変化することを確認します。データが多少グラグラするのはしかたないようです。5％の誤差があったとしても，12カウントも動くことになるのですから。スティックのトリムレバーは真ん中にセットしておきます。

スティックを最小値側に倒します。最小値カット幅調整用の半固定抵抗を回して，カウントが進む少し前にあわせておきます。次にスティックから手を離し，センター位置にします。この状態で読み取り値が80H近辺になるように発振周波数調整の半固定

表1　部品表

部品名	個数	部品名	個数
プロポ本体（フタバのFP-T2NBL）	1	**○カウンタ回路**	
○プロポ受信部		IC	
7kコイル（FCZ研究所の28MHz用でよい）	3	74LS08	1
半導体		74LS374	2
2SK241	1	74HC30	1
2SC845	1	74HC157	1
2SC1815-Y	1	74HC374	4
1N60	1	74HC4040	1
抵抗		抵抗	
1kΩ（茶黒赤）	1	10kΩ（茶黒橙）	18
10kΩ（茶黒橙）	1	（8個入りの集合抵抗を利用してもよい）	
100kΩ（茶黒黄）	1	コンデンサ	
コンデンサ		0.1μF（パスコン）	10
33pF	3	**○ホストインタフェイス回路**	
0.001μF	4	IC	
○受信波形処理回路		74HC02	1
IC		74HC14	1
74HC00	1	74HC74	1
74HC14	1	74HC123A	1
74HC74	2	74HC138	1
74HC123A	1	74HC161	1
74HC164	1	74HC163	1
半固定抵抗		74HC165	1
47kΩ	1	抵抗	
100kΩ	1	1.5kΩ（黒緑赤）	1
抵抗		コンデンサ	
56kΩ（緑青橙）	1	0.1μF（パスコン）	1+8
コンデンサ		ケース	1
0.001μ	1	スイッチ	10
0.047μ	1	D-SUBコネクタ（9ピンのメス）	1
0.1μ（パスコン）	1+5	その他，アンテナ接続用の端子，スペーサ，電線など	適宜

▶小生3月下旬にベルギーのブリュッセルに赴任することになりました。ブリュッセルでもOh！Xを読みたいと思い，海外送付を申し込みましたが，船便で年間1万920円，航空便で約2万5千円とその高さにびっくり。　脇本　眞也（35）X68000ACE-HD　千葉県

抵抗を調整します。スティックを値が最大になる側に倒して読み取り値がFF_Hになるかチェックします。ならないようなら発振周波数を引き上げ，今度はセンター位置で80_Hになるように最小値カット幅調整用半固定抵抗を調整します。これを何度か繰り返していい感触のところに追い込んでいきます。チャンネルごとの値のばらつきはプロポ側のトリムレバーで調整できます。

サイバースティック対応ソフトがあれば，実際に使ってみるのがよいでしょう。私は生まれて初めて買ったアクションゲームであるアフターバーナーが格好のターゲットでした。が……アフターバーナーをやると，両手がプロポでふさがるのでこのスイッチを操作するのはもうひとり呼んでくるか，足でやるしかないのですね(ハハハ)。

逃げ回るだけのアフターバーナーのむなしいこと……で，しかたなくフットスイッチを2つつけてみました。これは，快感です。線を伸ばさなくてはならないのはスイッチ，それもフットスイッチですから，椅子にふんぞり返ったままで遊べます。プロポを胸の前に構えて両手で操作しながら，足でミサイル発射！　終わった頃にはスネが筋肉痛でしたが……。

*　　*　　*

今回作ったのはコントローラ（プロポ）側が2chですが，簡単に4chにも対応できますのでサイバースティックでは幻となっているアナログチャンネルも生かすことができるというのは公然の秘密です。

回路自体はこれまで作ってきたものと比べるといくぶん大きなものになってしまいました。使ったICもフリップフロップ，ラッチ，カウンタ，ワンショット，シフトレジスタといった普通のANDやORといったものよりも少しややこしい機能を持ったもののオンパレードですし，動作も時間監視ありループありといった具合ですから，ハードに縁のなかった方はもちろん，デジタル回路の基礎を習ったくらいの方にとっても少々難しかったかもしれません（若干，反省)。

まあ，とにかくサイバースティックのインタフェイスを汎用のICの組み合わせで作ってしまうというもくろみは成功しました。幻のアナログチャンネル，お化けの12回目の4ビットデータなど，サイバースティックには隠された機能もたくさんありました。今回の回路のうちホストインタフェイスのところはそのままに，カウンタボードのところだけ作り変えればA/Dコンバータでもなんでもつなぎたい放題です。幻のチャンネルも有効に使えるようにできます。

すべてスイッチにすれば，なんと48個！16人同時プレイの対戦型テトリスも夢じゃない(?)。

さて，今度は何を作ろうか。

LSを使ったわけ

ほかがHCシリーズなのに，カウンタだけLSシリーズがあるので，妙な感じを持たれた方もいることでしょう。スイッチ周りをまじめにやるならCRのフィルタ＋シュミットゲート（HC14など）を使ってチャタリング（ON/OFF 時の信号のばたつき）を防止すべきなのです。それは百も承知なのですが，10ビット分というと，HC14が2パッケージ必要であり，今回のようにせせこましい基板では少々苦しいので，代わりにこの部分だけLSシリーズにして逃げたのです。

ここに普通の(入力にシュミット特性のない)C-MOSを使うと，入力がふらついたときにラッチアップを起こして過大電流が流れ，ICが壊れてしまう恐れがあります（もっとも，私自身はラッチアップが原因でICを壊した体験はありませんが)。

ハードウェアとソフトウェア

いままでソフト一辺倒で、デジタル回路にあまり馴染みのなかった方には回路図を読む前にちょっとした頭の切り換えが必要かもしれません。

ハードウェアの動作がソフトウェアと決定的に違うのは，ソフトウェアが基本的に1次元のシーケンシャルな流れであるのに対してハードウェアはカウンタならカウンタ，ゲートならゲートといった固められた機能単位が同時に動作し，互いに相手に信号を伝えあいつつ全体の回路というマクロな「物」の動作が形成される「並列/分散処理」が基本であるということです。今回のように○○回路といった具合にまとめられた部分をゲートなどよりも上位のモジュールと考えれば，回路間をつないでいる信号はメッセージと考えられるでしょう。

モジュール同士は同時に動きつつ，互いに相手にメッセージを送りあい，ひとつのラジコンスティックという機能を実現する……と考えれば，これはオブジェクト指向の一種と見なすこともできるのかもしれません。

目的の動作を得るための方法もかなり違うといえるでしょう。ソフトウェアでは目的を時間軸にそった流れで考えますが，ハードはある目的の動作を図形としてとらえるといったらよいでしょうか。「タイミング図」というのはその最たるものでしょう。デジタル回路では作り出される波形を元に新しい波形を作り，それをまたほかの波形と組み合わせて新しい波形を作り……，といったようにやっていくわけです(NZ-80KのPASIC：プロフェッショナル・ア・スケマティック・インストラクション・コード：……かもしれない)。

リスト1

```
1000 /*-----------------------------------------*/
1010 /*- ラジコン・スティックチェックプログラム  */
1020 /*-                                       -*/
1030 /*- 1990-02-11 written by M.kuwano        -*/
1040 /*-                                       -*/
1050 /*-   No rights reserved.                 -*/
1060 /*-                                       -*/
1070 /*-----------------------------------------*/
1080 char c(5)
1090 int stat,px,py,pz,pt
1100 px=0:py=0
1110 pr_info()
1120 astset()
1130 repeat
1140    stat = astick(c)
1150    disp_val(stat)
1160    disp_pos()
1170 until inkey$(0)=" "
1180 astrst()
1190 end
1200 func disp_val(stat;int)
1210    int i
1220    locate 0,9
1230    if stat<>0 then color 2: print"ERROR" else color 3:prin
t"READY"
1240    locate 0,8
1250    for i=0 to 3
1260        print right$("0"+hex$(c(i)),2);"  ";
1270    next
1280    i=&H80
1290    repeat
1300        pr_onoff(c(4) and i)
1310        i=i/2
1320    until i=0
1330    i=&H80
1340    repeat
1350        pr_onoff(c(5) and i)
1360        i=i/2
1370    until i=0
1380 endfunc
1390 func pr_onoff(s;int)
1400    if s=0 then print"ON   "; else print"OFF  ";
1410 endfunc
1420 func pr_info()
1430    screen 2,0,1,1
1440    console ,,0
1450    color 3
1460    locate 0,5:print"終了する時はスペースバーを押してください"
1470    color 1
1480    locate 0,7
1490    print"#1  #2  #3  #4  A+A' B+B'  C    D    E1   E2   F
  G   ?    ?    ?    ?    A    B    A'   B'"
1500     box(349,239,349+257,239+257,15)
1510     box(49,239,49+257,239+257,15)
1520 endfunc
1530 func disp_pos()
1540     line(350,240+py,350+255,240+py,0)
1550     line(350,240+c(0),350+255,240+c(0),15)
1560     py=c(0)
1570     line(350+px,240,350+px,240+255,0)
1580     line(350+c(1),240,350+c(1),240+255,15)
1590     px=c(1)
1600     line(50,240+pz,50+255,240+pz,0)
1610     line(50,240+c(2),50+255,240+c(2),15)
1620     pz=c(2)
1630     line(50+pt,240,50+pt,240+255,0)
1640     line(50+c(3),240,50+c(3),240+255,15)
1650     pt=c(3)
1660 endfunc
```

▶先日，友達と話をしているとき，「パイオニアの新製品はフルコンパチなんだよね」といわれた。そこで，僕はふと，「そうか，ソフトは100%の互換性があるのか」とマジに考えてしまった。バカだねー！　　西村　泰和（21）PC-9801RX2　千葉県

リスト2

```
==================  AJ.S  ==================
  1: *-----------------------------------------------------------------------
  2: *-      アナログ・ジョイスティック（サイバー・スティック）              -
  3: *-      読み込み関数                                                    -
  4: *-                                                                      -
  5: *-      1989-06-17 M.Kuwano, No rights reserved                         -
  6: *-      1990-02-11 デバッグ＆１２回転送対応に変更                       -
  7: *.......................................................................
  8: *-      呼び出し方（例）：                                              -
  9: *-      char c(5)                                                       -
 10: *-      astick(c)                                                       -
 11: *-      入るデータは・・                                                -
 12: *-              c(0).....右側スティック左右                             -
 13: *-              c(1).....右側スティック前後                             -
 14: *-              c(2).....左側スティック前後（スロットル）               -
 15: *-              c(3).....予備スティック（将来用？）                     -
 16: *-              c(4).....トリガ・ボタン                                 -
 17: *-              c(5).....トリガ・ボタン                                 -
 18: *-      と、なります                                                    -
 19: *-      配列は6バイト以上確保してください。                             -
 20: *-      配列のサイズチェックをやっていないので、あらぬところまで-
 21: *-      書き込んでしまいます。                                          -
 22: *.......................................................................
 23: *-      いつもながらインタプリタ／コンパイラ共用です                    -
 24: *-                                                                      -
 25: *-----------------------------------------------------------------------
 26:                 .include        doscall.mac
 27:                 .include        fdef.h
 28:                 .globl          _astick
 29:                 .globl          _astset
 30:                 .globl          _astrst
 31:
 32: IOCS            equ             $0f
 33:
 34: PPI_PORT_A      equ             $e9a001
 35: PPI_PORT_B      equ             $e9a003
 36: PPI_PORT_C      equ             $e9a005
 37: PPI_CWR         equ             $e9a007
 38:
 39: RQ_ASSERT       equ             $8
 40: RQ_NEGATE       equ             $9
 41:
 42: TIME_LIMIT1     equ             1000
 43: TIME_LIMIT2     equ             100
 44:                 .text
 45:                 .even
 46: *
 47: *  インフォメーション・テーブル
 48: *
 49: *
 50:                 .dc.l           AS_INIT
 51:                 .dc.l           AS_RUN
 52:                 .dc.l           AS_END
 53:                 .dc.l           AS_SYS
 54:                 .dc.l           AS_BRK
 55:                 .dc.l           AS_CTRL_D
 56:                 .dc.l           AS_RES1
 57:                 .dc.l           AS_RES2
 58:                 .dc.l           PTR_TOKEN
 59:                 .dc.l           PTR_PARAM
 60:                 .dc.l           PTR_EXEC
 61:                 .dc.l           0,0,0,0,0
 62:
 63: AS_RES1:
 64: AS_RES2:
 65: AS_END:
 66: AS_BRK:
 67: AS_CTRL_D:
 68: AS_INIT:
 69: AS_RUN:
 70: AS_SYS:
 71:                 rts
 72:
 73: *
 74: *  トークン・テーブル
 75: *
 76: PTR_TOKEN:
 77:                 .dc.b           'astick',0
 78:                 .dc.b           'astset',0
 79:                 .dc.b           'astrst',0
 80:                 .dc.b           0
 81:                 .even
 82: *
 83: *  パラメータ・テーブル
 84: *
 85: PTR_PARAM:
 86:                 .dc.l           ASTICK_PAR
 87:                 .dc.l           ASTSET_PAR
 88:                 .dc.l           ASTRST_PAR
 89:
 90: *
 91: *  パラメータＩＤテーブル
 92: *
 93: ASTICK_PAR:
 94:                 .dc.w           ary1_c
 95:                 .dc.w           int_ret
 96: ASTSET_PAR:
 97:                 .dc.w           void_ret
 98: ASTRST_PAR:
 99:                 .dc.w           void_ret
100:
101: *
102: *  関数アドレステーブル
103: *
104: PTR_EXEC:
105:                 .dc.l           astick
```


▶中嶋負けるな！　亜久里ガンバレ！　去年は古舘伊知郎の解説がうるさかったけど，今年はそうでないことを祈る。
小川　恭孝 (18) X1turboZ 千葉県

```
106:                .dc.l           _astset
107:                .dc.l           _astrst
108:
109: *
110: *  スタック・バッファ
111: *
112: SPBUF:
113:                .ds.l           1
114:
115:                .even
116: *
117: *  アナログ・ジョイスティック読みだし(インタプリタ用)
118: *
119: astick:
120:                movea.l         12(sp),a1
121:                lea             10(a1),a1
122:                move.l          a1,-(sp)
123:                bsr             _astick
124:                addq.l          #4,sp
125:                moveq.l         #0,d0
126:                rts
127:
128: *
129: *  アナログ・ジョイスティック用に、PC4を1にする
130: *
131: _astset:
132:                clr.l           -(sp)                   *     SPBUF = _SUPER(0);
133:                dc.w            _SUPER
134:                addq.l          #4,sp
135:                move.l          d0,SPBUF
136:                movea.l         #PPI_CWR,a0             *     *ppi_cwr = RQ_NEGATE;
137:                move.b          #RQ_NEGATE,(a0)
138:                move.l          SPBUF,d0                *     _SUPER(SPBUF);
139:                bmi             astset_already_super
140:                move.l          d1,-(sp)
141:                dc.l            _SUPER
142:                addq.l          #4,sp
143:        astset_already_super:
144:                moveq.l         #0,d0
145:                lea.l           AS_RETVAL,a0
146:                move.w          d0,2(a0)
147:                rts
148:
149: *
150: *  プログラム終了後、PC4を0に戻しておかないと
151: *  デジタル・モード用のソフトのうち動かなくなるも
152: *  のがでてくるらしい
153: *
154: _astrst:
155:                clr.l           -(sp)                   *     SPBUF = _SUPER(0);
156:                dc.w            _SUPER
157:                addq.l          #4,sp
158:                move.l          d0,SPBUF
159:                movea.l         #PPI_CWR,a0             *     *ppi_cwr = RQ_ASSERT;
160:                move.b          #RQ_ASSERT,(a0)
161:                move.l          SPBUF,d0                *     _SUPER(SPBUF);
162:                bmi             astrst_already_super
163:                move.l          d1,-(sp)
164:                dc.l            _SUPER
165:                addq.l          #4,sp
166:        astrst_already_super:
167:                moveq.l         #0,d0
168:                lea.l           AS_RETVAL,a0
169:                move.w          d0,2(a0)
170:                rts
171:
172: *
173: *  アナログ・ジョイスティック読みだし(コンパイル時用)
174: *
175: _astick:
176:                bsr             get_astick              * get_astick();
177:                lea.l           AS_RETVAL,a0
178:                move.l          d0,6(a0)
179:                move.l          4(sp),d0
180:                move.l          d0,-(sp)
181:                bsr             aj_compile
182:                addq.l          #4,sp
183:                lea.l           AS_RETVAL,a0
184:                move.l          6(a0),d0
185:                rts
186:
187:
188: *
189: * アナログ・ジョイスティックデータ取り込み
190: *
191: * a0 Buffer_pointer
192: * a1 PPI_PORT_A
193: * a2 PPI_CWR
194: * d0 data
195: * d1 data
196: * d2 Loop counter
197: * d3 Timeout counter
198: *
199: get_astick:                                             * get_astick() {
200:                clr.l           -(sp)                   *     SPBUF = _SUPER(0);
201:                dc.w            _SUPER
202:                addq.l          #4,sp
203:                move.l          d0,SPBUF
204:                move.w          sr,-(sp)                *     PUSH(SR);
205:                ori.w           #$0700,sr               *     disable_trap();
206:                lea.l           AS_TMP_BUF,a0           *     buffer_pointer = AS_TMP_BUF;
207:                movea.l         #PPI_PORT_A,a1          *     ppi_port_a     = PPI_PORT_A;
208:                movea.l         #PPI_CWR,a2             *     ppi_cwr        = PPI_CWR;
209:
210:                move.w          #5,d2                   *     loop_counter = 5;
211:                moveq.l         #0,d0                   *     joydata = 0;
212:                move.w          #TIME_LIMIT1,d3         *     timer = TIME_LIMIT1
213:        _astick_0:                                      *     do {
```

▶表紙をパッと見たときはただまわりが白くなっただけだと思いましたが，よく見ると完全に変わっており，気に入りました。
三井田　孝欧（17）X68000ACE，X1/turboZ III　東京都

```
214:            move.b      #RQ_ASSERT,(a2)     *       *ppi_cwr = RQ_ASSERT
215:        _astick_1:
216:            move.b      (a1),d0             *       while (((data = *ppi_port_a & 0x60) != 0) && timer--)
217:            move.b      d0,d1               *           ;
218:            andi.b      #$60,d0
219:            dbeq        d3,_astick_1
220:            bne         _astick_timeout     *       if (!timer) goto _astick_timeout;
221:            move.b      d1,(a0)+            *       *buffer_pointer++ = data;
222:            move.b      #RQ_NEGATE,(a2)     *       *ppi_cwr = RQ_NEGATE;
223:        _astick_11:
224:            btst.b      #5,(a1)             *        while(!(*ppi_port_a & 0x20) && timer--)
225:            dbne        d3,_astick_11       *           ;
226:            beq         _astick_timeout     *        if (!timer)   goto _astick_timeout;
227:        _astick_12:                         *        while(((data = *ppi_port_a) & 0x40) && timer--)
228:            move.b      (a1),d1
229:            btst        #6,d1               *           ;
230:            dbeq        d3,_astick_12
231:            bne         _astick_timeout     *        if (!timer) goto _astick_timeout;
232:
233:            move.b      d1,(a0)+            *        *buffer_pointer++ = data;
234:            dbra        d2,_astick_0        *   } while (loop_counter--);
235:
236:
237:            moveq.l     #0,d3               *    retstat = 0;
238:        _astick_exit:
239:            move.w      (sp)+,sr            *    sr = POP(); /* 割り込みフラグを戻す        */
240:            move.b      #RQ_NEGATE,(a2)     *    *ppi_cwr = RQ_NEGATE;
241:            move.l      SPBUF,d1            *    _SUPER(SPBUF);
242:            bmi         _astick_already_super
243:            move.l      d1,-(sp)
244:            dc.l        _SUPER
245:            addq.l      #4,sp
246:        _astick_already_super:
247:            move.l      d3,d0
248:            rts                             * }
249:
250:        _astick_timeout:
251:            moveq.l     #1,d3               *    data = 1;
252:            bra         _astick_exit
253: *
254: *  アナログ・ジョイスティックデータ編集
255: *
256:
257: aj_compile:                           * aj_compile(pack_data) {
258:            movea.l     4(sp),a0
259:            lea.l       AS_TMP_BUF,a1
260:
261:            move.b      2(a1),d1      *    pack_data[0] = joy_raw_data[6] | (joy_raw_data[2] << 4);
262:            andi.b      #$f,d1
263:            lsl         #4,d1
264:            move.b      6(a1),d2
265:            andi.b      #$f,d2
266:            or.b        d2,d1
267:            move.b      d1,(a0)+
268:
269:            move.b      3(a1),d1      *    pack_data[1] = joy_raw_data[7] | (joy_raw_data[3] << 4);
270:            andi.b      #$f,d1
271:            lsl         #4,d1
272:            move.b      7(a1),d2
273:            andi.b      #$f,d2
274:            or.b        d2,d1
275:            move.b      d1,(a0)+
276:
277:            move.b      4(a1),d1      *    pack_data[2] = joy_raw_data[8] | (joy_raw_data[4] << 4);
278:            andi.b      #$f,d1
279:            lsl         #4,d1
280:            move.b      8(a1),d2
281:            andi.b      #$f,d2
282:            or.b        d2,d1
283:            move.b      d1,(a0)+
284:
285:            move.b      5(a1),d1      *    pack_data[3] = joy_raw_data[9] | (joy_raw_data[5] << 4);
286:            andi.b      #$f,d1
287:            lsl         #4,d1
288:            move.b      9(a1),d2
289:            andi.b      #$f,d2
290:            or.b        d2,d1
291:            move.b      d1,(a0)+
292:
293:            move.b      0(a1),d1      *    pack_data[4] = joy_raw_data[1] | (joy_raw_data[0] << 4);
294:            andi.b      #$f,d1
295:            lsl         #4,d1
296:            move.b      1(a1),d2
297:            andi.b      #$f,d2
298:            or.b        d2,d1
299:            move.b      d1,(a0)+
300:
301:            move.b      11(a1),d1     *    pack_data[5] = joy_raw_data[10] | (joy_raw_data[11] << 4);
302:            andi.b      #$f,d1
303:            lsl         #4,d1
304:            move.b      10(a1),d2
305:            andi.b      #$f,d2
306:            or.b        d2,d1
307:            move.b      d1,(a0)+
308:
309:            rts                       * }
310:
311:            .even
312: AS_TMP_BUF:
313:            .ds.b       12
314: AS_ACK_BUF:
315:            .ds.b       5
316:            .even
317: AS_RETVAL:
318:            .dc.w       0
319:            .dc.l       0
320:            .dc.l       0
321:            .end
```


▶つ，ついに20歳になってしまった。これでもう悪事ができなくなった。もう少年Aではなくなるのは少し悲しい。　池田　和繁（20）X1/turbo　東京都

CARD.FNC

Mounai Toshiyuki 毛内 俊行

ばばぬき，神経衰弱，ポーカーからソリティア，占いまで。たとえばトランプなら１組のカードで数100種のゲームができる。カードゲームとは実に奥が深い。X-BASICで美しいカード表示を手軽に実現する外部関数をカードデータとともに紹介しよう。

トランプや花札，UNOなどのカードゲームといえば，皆さんもパーティなどで大勢の人が集まったときに遊んだ記憶があるでしょう。そして，そのカードゲームをパソコンで作ってみようと思ったことだってあるのではないでしょうか。

ところが，カードゲームを作る場合とても厄介なことがひとつあります。それはカードのパターンデータを作らなければいけないことです。この作業はヘタをするとプログラム本体を作るより大変な作業なのです。そこで，この厄介な作業を少しでも楽にしようと考え，カードの定義，表示を行うためのプログラムCARD.FNC（以下CARDと呼ぶ）を作りました。

CARDの特徴

CARDは，カードゲームを作る人にとって大変便利なプログラムです。それではいったいCARDがどのように便利なものなのかを説明しましょう。

1） BASIC上で手軽に使える

名前からもわかるように，CARDはX-BASICの外部関数として作られています。そのため，BASICに一度組み込んでしまえば，命令ひとつで自由にカードを表示することができるのです。いままでなら，カードパターンを作るプログラムを用意しなくてはならなかったのですから，これはもう嬉しい限りです。

2） カードパターンの定義が自由にできる

CARDにはあらかじめ，トランプのパターンが合計54枚定義されています。しかし，ほかのカードゲームのカードだってもちろん使うことができます。そのため，CARDにはパターンを定義する命令があらかじめ用意されています。なんとCARDは最大60枚ものパターンの定義が可能です。

3） カードの絵が綺麗である

自分でいうと結構あつかましく聞こえますが，これは私の自慢だったりします。いままでトランプのゲームは，いろいろ紹介されていますし，市販されているものもたくさんあります。しかし残念なことに，私が見る限りではそれらのソフトのほとんどは，カードのデザインが省略されて結構見苦しいものが多いのです。

CARDは，48×96ドットという比較的大きなサイズのカードを用意しているので，トランプの図柄も従来よりは細かく綺麗なものが用意できたと思います。また自分でカードを作る際も，よほど細かい図柄でない限り自由に作ることができるでしょう。

以上の３つがCARDのセールスポイント，つまりウリ文句です。しかし悲しいことに，おいしい話には裏があります。長所だけでなく短所だってあるのです。やはり長所だけ紹介しては公平じゃないので，CARDの持つ短所も一応述べておきます。

1） 画面モードに制限がある

CARDは画面モードが16色のときでないと使えません。256色，65536色モードのときはエラー警告をします。ただし，画面サイズは768×512～256×256までどのサイズでも大丈夫です。とりあえず16色に対応していれば十分と判断したので，それ以外のモードのことは考えませんでした。我慢してください。

2） プログラムのサイズが大きい

CARDは，データの汎用性を重視したため，圧縮などの操作は一切行いませんでした。そのために実際のCARD本体の大きさは，なんと135Kバイトにもなってしまいました（皆さんが入力するデータは，思いっきり圧縮してあるので心配しないように）。でもやはり，メモリを増設していないマシンでは多少辛いものがあります。特にOPMAやRAMディスクのようなメモリ喰いと同居するのは大変でしょう。やはりユーザーメモリが1Mバイトの人は，RAMの増設は必須科目ですね。

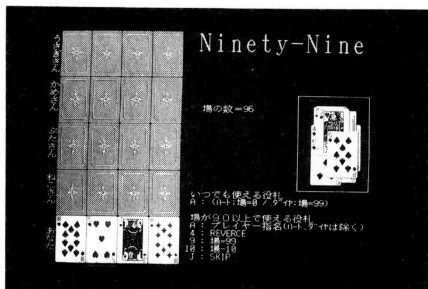
このように短所もありますが，命令ひとつで，カードを自由に操作できるのですから，これを使うのと使わないのとでは，ゲームを作るのに天と地ほどの差があります。だまされたと思ってプログラムを入力してみてください。

プログラムの入力

プログラムは２つあります。まず，リスト１をBASICから入力してください。入力が終わったら，リスト２をマシン語入力ツールを使って入力しましょう。リスト２のファイル名は必ず「FONT.DAT」とし

裏面も変更できる

絵札も美しい

▶ダンジョンマスターで，ウーヤーターと呪文を唱えると，画面が揺れて，敵に大きなダメージを与えられます(ウソ)。　太田　敬三（20）X68000PRO　東京都

てください。リスト2の入力がすんだら，ディスクにセーブされているのを確認します。確認したらリスト1を実行してください。リスト2のデータを展開して，自動的にメインプログラムを作成してくれます。

エラーが発生しなければ，十数分で展開が終了します。ディレクトリの中に「CARD.FNC」というファイルがあればOKです。完成したらBASICのディレクトリにコピーして，BASIC.CNFの中に，FUNC=CARDと書き足しましょう。これで次回からは，X-BASICを起動するだけで，CARDが使えます。

もし，展開中にエラーが発生した場合は，一応キーボードからfcloseall( )を実行してください。まず心配はないと思いますが，ファイルがオープンされたままでは，最悪の場合ディスクを壊すおそれがあります(確かBASICでは大丈夫だと聞いていましたが，私はRAMディスクを壊しました)。万一を考えて，リスト1とリスト2は，ほかのディスクにバックアップを取っておいたほうがいいでしょう。

また，プログラムの展開中にいくつかデータファイルを作成するので，作成するディスクは300Kバイトほどの空きエリアを確保しておいてください。これがなくてもエラーの原因になります。

CARDの使い方

CARDを装備したX-BASICには3つの関数が新たに使えます。以下にその関数と機能を紹介します。

●c_put(x,y,n)

座標(x,y)で指定したグラフィック画面に，カード番号nで指定したカードを表示します。パラメータの形式はすべてintです。戻り値はありません。カード番号nは，0～59の範囲で指定してください。カード番号についてはあとで説明します。

●c_get(n,ca)

カード番号nのパターンを配列caに読み込みます。画面上のパターンの大きさは48×96ドットなので，配列の大きさは2304バイト必要です。パラメータ形式はnがint，caはcharです。なお，caに格納されたデータは2ドット＝1バイトで単純に横方向に並んだデータなので，16色モードのときのBASICのput命令のデータと互換性があります。

●c_set(n,ca)

カード番号nに配列caに定義されたパターンをセットします。つまりc_getと反対の動作をします。パターンやパラメータの形式はc_getと同じです。

以上が追加される関数です。たったこれだけ？　と思う人もいるかもしれませんが，これだけあれば，ひととおりのことが簡単にできるのです。この機能を少ないと思うか多いと思うか，とりあえず使ってから考えましょう。

トランプのパターンを使う場合

CARDにはあらかじめ，トランプのパターンが用意されています。本来，トランプを使うときは「スペードの10」とか「クラブのK」とかいうように，スートと数によって識別します。ところがCARDではほかのカードを使うときのために汎用性を持たせ，カード番号という概念を取り入れています。これでは理解しにくいので，カード番号とカードの関係を説明しておきましょう。

ここではトランプのスート（スペードとかハートなんかのマークのこと）にスート番号という数を新たに考えます。スート番号は以下のとおりです。

スペード＝0
ハート　＝1
ダイヤ　＝2
クラブ　＝3

余談ですが，このスートの順番はまったくのでたらめではなく，コントラクトブリッジやナポレオンの，スートの優先順位に準じているという由緒正しいものですので，ちゃんと覚えておきましょう。

さて本題に入りますが，このスート番号を使うと，目的のカードのカード番号は次の数式で表すことができます。

カード番号＝スート番号×13＋数

つまり，スペードのAを表示したければ，

カード番号＝0×13＋1＝1

となりますし，ダイヤの8を表示したければ，

カード番号＝2×13＋8＝34

となるわけです。

さて，しかしこれでは表示できないカードがあります。それはカード番号0の，カードの裏のパターンと，カード番号53のジョーカーです。しかしこれらのカードは，存在が特殊なために，私が「こうしなさい」というわけにはいきませんので，皆さんで独自に扱ってください。とりあえず，

```
c_put(20,150,0)
c_put(50,50,53)
```

のように，カード番号を直接扱うのが楽だろうと思います。

最後に

1989年11月号で「ばばぬき」を発表して以来，このような外部関数があったらいいなと思い，ついに自分で作ってしまいました。実際に作ってみると，意外と簡単で(そりゃグラフィックパターンのPUTだけだから当たり前なのだけど）結構短い時間でプログラムは完成してしまいました。

ところが完成してびっくり。なんとプログラムの大きさが135Kバイトもあるではありませんか！　ページの関係と，入力する人の体力を考えれば，こんな化け物サイズのプログラムをそのまま掲載するわけにはいかないのは明らかです。やがて，編集室から「データを5Kバイトまでに圧縮しなさーい」という，かぐや姫も真っ青，一休さんもびっくりの無理難題を押しつけられてしまい，思わず「ひえぇー」(ここでムンクの「叫び」のポーズ）と頭を抱えてしまったのでした。

まあ編集の人からのアドバイスもあったりして，いろいろな方法を試みるうちに5Kバイトとまではいかないにしても，それに近いサイズにたどりつくことができました。しかしそのあいだに結構あくどいこともやっているので，展開プログラムや圧縮データの解析はあまりお勧めできません。CARD本体のプログラムについては，データ部を除いた部分のソースリストを載せましたので，こちらのほうはどうぞ参考にしてください。

そんなわけで，苦労の多かったこのプログラムも，皆さんに喜んでもらえれば私としても嬉しい限りです。なお，特集ではCARDを使ったゲームが紹介されていますので，そちらも参考にしてください。

▶3月号の特集は，初心者にとって，音楽への「NICE MIDI PASS」ですね(ギャグがさえないなぁ)。　小笠原　洋（15）X68000ACE-HD，X1G　東京都

リスト1 CARD.S

```
==================  CARD.S  ==================
    1: *================================
    2: *
    3: *        X-BASIC
    4: *        カード表示用外部関数
    5: *
    6: *        c_put(x,y,n)       へ へ
    7: *        c_get(n,cr)        の の
    8: *        c_set(n,cr)         も
    9: *                            へ
   10: *================================
   11: *
   12: *=======================
   13: *          定数定義
   14: *=======================
   15: *
   16: XMAX:      equ     47                 *横-1
   17: YMAX:      equ     95                 *縦-1
   18: MMAX:      equ     2304               *横×縦
   19: CMAX:      equ     60                 *キャラクタ総数
   20: *
   21: *=======================
   22: *    INFORMATION TABLE
   23: *=======================
   24: *
   25:    .dc.l   F_INT      *起動時初期化ルーチンのアドレス
   26:    .dc.l   F_RUN      *RUN 実行時のエントリアドレス
   27:    .dc.l   F_END      *END 実行時のエントリアドレス
   28:    .dc.l   F_SYSTEM   *SYSTEM/EXIT 実行時のエントリアドレス
   29:    .dc.l   F_BRK      *BREAK/ctrl-C実行時のエントリアドレス
   30:    .dc.l   F_CTRL_D   *1行入力中のctrl-D入力時のエントリアドレス
   31:    .dc.l   F_DMY1     *リザーブ
   32:    .dc.l   F_DMY2     *リザーブ
   33:    .dc.l   F_TOKEN    *トークンテーブルの先頭アドレス
   34:    .dc.l   F_PARAM    *パラメータの実行アドレス
   35:    .dc.l   F_EXEC     *実行アドレステーブルの先頭アドレス
   36:    .dc.l   0,0,0,0,0  *空白20バイト(予備)
   37: *
   38: *=======================
   39: *         SUB ENTRY
   40: *=======================
   41: *
   42: F_INT:
   43: F_RUN:
   44: F_END:
   45: F_SYSTEM:
   46: F_BRK:
   47: F_CTRL_D:
   48: F_DMY1:
   49: F_DMY2:
   50:    rts
   51: *
   52: *=======================
   53: *         マクロ定義
   54: *=======================
   55: *
   56: IOCS       macro   nm
   57:    move.l  #nm,d0
   58:    trap    #15
   59:    endm
   60: *
   61: *=======================
   62: *      TOKEN TABLE
   63: *=======================
   64: *
   65: F_TOKEN:
   66:    .dc.b   'c_put',0
   67:    .dc.b   'c_get',0
   68:    .dc.b   'c_set',0
   69:    .dc.b   0
   70:    .even
   71: *
   72: *=======================
   73: *     PARAMETER TABLE
   74: *=======================
   75: *
   76: F_PARAM:
   77:    .dc.l   PUT_TBL
   78:    .dc.l   GET_TBL
   79:    .dc.l   SET_TBL
   80: *
   81: *=======================
   82: *       EXEC TABLE
   83: *=======================
   84: *
   85: F_EXEC:
   86:    .dc.l   PUT_ENT
   87:    .dc.l   GET_ENT
   88:    .dc.l   SET_ENT
   89: *
   90: *=======================
   91: *   PARAMETER ID TABLE
   92: *=======================
   93: *
   94: PUT_TBL:
   95:    .dc.w   $0002    *画面座標X    int
   96:    .dc.w   $0002    *画面座標Y    int
   97:    .dc.w   $0002    *カード番号    int
   98:    .dc.w   $ffff    *戻り値         VOID
   99: GET_TBL:
  100:    .dc.w   $0002    *カード番号    int
  101:    .dc.w   $0034    *カードデータ char,1次元配列
  102:    .dc.w   $ffff    *戻り値         VOID
  103: SET_TBL:
  104:    .dc.w   $0002    *各パラメータは、
  105:    .dc.w   $0034    *上の関数(c_get)
  106:    .dc.w   $ffff    *のものと同じ。
  107: *
  108: *=======================
  109: *      MAIN PROGRAM
  110: *=======================
  111: *
  112: PUT_ENT:
  113:    lea.l   DAT,a1
  114:    move.l  12(sp),d1
  115:    move.w  d1,0(a1)
  116:    move.l  22(sp),d2
  117:    move.w  d2,2(a1)
  118:    add.w   #XMAX,d1
  119:    add.w   #YMAX,d2
  120:    move.w  d1,4(a1)
  121:    move.w  d2,6(a1)
  122: *
  123:    move.l  32(sp),d2
  124:    cmp.l   #CMAX,d2
  125:    bcc     ERROR2
  126:    addq.l  #1,d2
  127:    bsr     GTVADR
  128: *
  129:    move.l  d1,8(a1)
  130:    add.l   #MMAX-1,d1
  131:    move.l  d1,12(a1)
  132:    IOCS    $bf
  133: *
  134:    tst.l   d0
  135:    bne     ERROR1
  136: *
  137:    move.l  #0,d0
  138:    rts
  139: *
  140: GET_ENT:
  141:    move.l  12(sp),d2
  142:    cmp.l   #CMAX,d2
  143:    bcc     ERROR2
  144:    addq.l  #1,d2
  145:    bsr     GTVADR
  146: *
  147:    move.l  #2303,d0
  148:    move.l  d1,a0
  149:    move.l  22(sp),a1
  150: *
  151:    move.w  8(a1),d0
  152:    cmp.w   #$8ff,d0
  153:    bcs     ERROR3
  154:    adda.l  #10,a1
  155: G1:
  156:    move.b  (a0)+,(a1)+
  157:    dbra    d0,G1
  158:    move.l  #0,d0
  159:    rts
  160: *
  161: SET_ENT:
  162:    move.l  12(sp),d2
  163:    cmp.l   #CMAX,d2
  164:    bcc     ERROR2
  165:    addq.l  #1,d2
  166:    bsr     GTVADR
  167: *
  168:    move.l  #2303,d0
  169:    move.l  d1,a0
  170:    move.l  22(sp),a1
  171: *
  172:    move.w  8(a1),d0
  173:    cmp.w   #$8ff,d0
  174:    bcs     ERROR3
  175:    adda.l  #10,a1
  176: S1:
  177:    move.b  (a1)+,(a0)+
  178:    dbra    d0,S1
  179:    move.l  #0,d0
  180:    rts
  181: *
  182: GTVADR:
  183:    move.l  #PAT,d1
  184:    sub.l   #MMAX*2,d1
  185: GVLP:      add.l   #MMAX,d1
  186:    dbra    d2,GVLP
  187:    rts
  188: *
  189: *=======================
  190: *          ERROR
  191: *=======================
  192: *
  193: ERROR1:
  194:    lea.l   ER1MES,a1
  195:    bra     ERROR
  196: ERROR2:
  197:    lea.l   ER2MES,a1
  198:    bra     ERROR
  199: ERROR3:    lea.l   ER3MES,a1
  200: *
  201: ERROR:
  202:    move.l  #1,d0
  203:    rts
  204: *
  205: *=======================
  206: *         DATA AREA
  207: *=======================
  208: *
  209: DAT:        .dc.w   0,0,0,0
  210:    .dc.l   0,0
  211: ER1MES:     .dc.b   '画面が初期化されていないか、座標'
  212: ER2MES:     .dc.b   'パラメータに誤りがあります'
  213:    .dc.b   10,13,0
  214: ER3MES:     .dc.b   '配列の変数の大きさが足りません'
  215:    .dc.b   10,13,0
  216: *
  217: PAT:
```

▶関係ないけど，女性がハイライトを吸う光景って，麻雀してるより怖い！（私は見てしまった)。これは偏見でしょうか？　　菅野　弘治（17）X68000ACE，X1F　東京都

リスト2 FONT.DAT

```
0000  48 55 00 00 00 00 00 00  : 9D
0008  00 00 00 00 00 02 1D EA  : 09
0010  00 00 00 00 00 00 00 00  : 00
0018  00 00 00 2C 00 00 00 00  : 2C
0020  00 00 00 00 00 00 00 00  : 00
0028  00 00 00 00 00 00 00 00  : 00
0030  00 00 00 00 00 00 00 00  : 00
0038  00 00 00 00 00 00 00 00  : 00
0040  00 00 00 40 00 00 00 40  : 80
0048  00 00 00 40 00 00 00 40  : 80
0050  00 00 00 40 00 00 00 40  : 80
0058  00 00 00 40 00 00 00 40  : 80
0060  00 00 00 42 00 00 00 56  : 98
0068  00 00 00 62 00 00 00 00  : 62
0070  00 00 00 00 00 00 00 00  : 00
0078  00 00 00 00 00 00 00 00  : 00
---------------------------------
SUM:  48 55 00 D0 00 02 1D 40  DD0A

0080  4E 75 63 5F 70 75 74 00  : DE
0088  63 5F 67 65 74 00 63 5F  : C4
0090  73 65 74 00 00 00 00 00  : 4C
0098  00 6E 00 00 00 76 00 00  : E4
00A0  00 7C 00 00 00 82 00 00  : FE
00A8  00 DC 00 00 01 14 00 02  : F3
00B0  00 02 00 02 FF FF 00 02  : 04
00B8  00 34 FF FF 00 02 00 34  : 68
00C0  FF FF 43 F9 00 00 01 7C  : B7
00C8  22 2F 00 0C 33 41 00 00  : D1
00D0  24 2F 00 16 33 42 00 02  : E0
00D8  D2 7C 00 2F D4 7C 00 5F  : 2C
00E0  33 41 00 04 33 42 00 06  : F3
00E8  24 2F 00 20 B4 BC 00 00  : E3
00F0  00 3C 64 00 00 B6 52 82  : 2A
00F8  61 00 00 90 23 41 00 08  : 5D
---------------------------------
SUM:  F3 BA E4 C3 28 76 2A 04  9DD1

0100  D2 BC 00 00 08 FF 23 41  : F9
0108  00 0C 20 3C 00 00 00 BF  : 27
0110  4E 4F 4A 80 66 00 00 8C  : 59
0118  70 00 4E 75 24 2F 00 0C  : 92
0120  B4 BC 00 00 00 3C 64 00  : 10
0128  00 82 52 82 61 5C 20 3C  : 6F
0130  00 00 08 FF 20 41 22 6F  : F9
0138  00 16 30 29 00 08 B0 7C  : A3
0140  08 FF 65 6E D3 FC 00 00  : A9
0148  00 0A 12 D8 51 C8 FF FC  : 08
0150  70 00 4E 75 24 2F 00 0C  : 92
0158  B4 BC 00 00 00 3C 64 4A  : 5A
0160  52 82 61 26 20 3C 00 00  : B7
0168  08 FF 20 41 22 6F 00 16  : 0F
0170  30 29 00 08 B0 7C 08 FF  : 94
0178  65 38 D3 FC 00 00 00 0A  : 76
---------------------------------
SUM:  5F 12 5B 01 4D 65 E4 30  40D4

0180  10 D9 51 C8 FF FC 70 00  : 6D
0188  4E 75 22 3C 00 00 01 EA  : 0C
0190  92 BC 00 00 12 00 D2 BC  : EE
0198  00 00 09 00 51 CA FF F8  : 1B
01A0  4E 75 43 F9 00 00 01 8C  : 8C
01A8  60 0E 43 F9 00 00 01 AC  : 57
01B0  60 06 43 F9 00 00 01 C9  : 6C
01B8  70 01 4E 75 00 00 00 00  : 34
01C0  00 00 00 00 00 00 00 00  : 00
01C8  00 00 00 00 89 E6 96 CA  : CF
01D0  82 AA 8F 89 8A FA 89 BB  : 0C
01D8  82 B3 82 EA 82 C4 82 A2  : 0B
01E0  82 C8 82 A2 82 A9 81 41  : 5B
01E8  8D C0 95 57 83 70 83 89  : 38
01F0  83 81 81 5B 83 5E 82 C9  : 0C
01F8  8C EB 82 E8 82 AA 82 A0  : 2F
---------------------------------
SUM:  90 E5 BE 13 01 8B EE F9  B234

0200  82 E8 82 DC 82 B7 0A 0D  : 18
0208  00 94 7A 97 F1 82 CC 95  : 79
0210  CF 90 94 82 CC 91 E5 82  : 39
0218  AB 82 B3 82 AA 91 AB 82  : CA
0220  E8 82 DC 82 B9 82 F1 0A  : FE
0228  0D 00 00 00 F7 FF 00 00  : 03
0230  F7 FF 00 00 E3 FF 00 00  : D8
0238  E3 FF 00 00 C1 FF 00 00  : A2
0240  C1 FF 00 00 80 FF 00 00  : 3F
0248  80 FF 00 00 00 7F 00 00  : FE
0250  00 7F 00 00 00 7F 00 00  : FE
0258  00 7F 00 00 94 FF 00 00  : 12
0260  E3 FF 00 00 C1 FF 00 00  : A2
0268  88 FF 00 00 88 FF 00 00  : 0E
0270  00 7F 00 00 00 7F 00 00  : FE
0278  00 7F 00 00 00 7F 00 00  : FE
---------------------------------
SUM:  77 06 1F F9 9A D2 57 B0  0C77

0280  00 7F 00 00 00 7F 00 00  : FE
0288  80 FF 00 00 80 FF 00 00  : FE
0290  C1 FF 00 00 C1 FF 00 00  : 80
0298  E3 FF 00 00 F7 FF 00 00  : D8
02A0  F7 FF 00 00 F7 FF 00 00  : EC
```

```
02A8  F7 FF 00 00 F7 FF 00 00  : EC
02B0  E3 FF 00 00 E3 FF 00 00  : C4
02B8  C1 FF 00 00 80 FF 00 00  : 3F
02C0  00 7F 00 00 80 FF 00 00  : FE
02C8  C1 FF 00 00 E3 FF 00 00  : A2
02D0  E3 FF 00 00 F7 FF 00 00  : D8
02D8  F7 FF 00 00 F7 FF 00 00  : EC
02E0  E3 FF 00 00 C1 FF 00 00  : A2
02E8  C1 FF 00 00 C1 FF 00 00  : 80
02F0  C1 FF 00 00 A2 FF 00 00  : 61
02F8  00 7F 00 00 00 7F 00 00  : FE
---------------------------------
SUM:  B6 70 00 00 FE F0 00 00  6965

0300  00 7F 00 00 00 7F 00 00  : FE
0308  00 7F 00 00 94 FF 00 00  : 12
0310  F7 FF 00 00 E3 FF 00 00  : D8
0318  C1 FF 00 00 DF AF 00 00  : 4E
0320  8F 07 00 00 07 07 00 00  : A4
0328  07 07 00 00 07 07 00 00  : 1C
0330  07 07 00 00 57 8F 00 00  : F4
0338  8F DF 00 00 DF 8F 00 00  : DC
0340  DF 8F 00 00 8F 07 00 00  : 04
0348  07 07 00 00 07 07 00 00  : 1C
0350  8F 07 00 00 DF DF 00 00  : 54
0358  DF 8F 00 00 DF 8F 00 00  : DC
0360  DF 77 00 00 DF 77 00 00  : AC
0368  AF F7 00 00 AF EF 00 00  : 44
0370  AF DF 00 00 77 BF 00 00  : C4
0378  07 7F 00 00 77 77 00 00  : 74
---------------------------------
SUM:  7C E8 00 00 6A 70 00 00  C32D

0380  77 07 00 00 07 EF 00 00  : 74
0388  F7 EF 00 00 EF CF 00 00  : A4
0390  DF CF 00 00 8F AF 00 00  : EC
0398  77 AF 00 00 F7 6F 00 00  : 8C
03A0  F7 07 00 00 77 EF 00 00  : 64
03A8  8F EF 00 00 07 8F 00 00  : 14
03B0  7F 77 00 00 7F 7F 00 00  : F4
03B8  7F 7F 00 00 0F 0F 00 00  : 1C
03C0  F7 77 00 00 F7 77 00 00  : DC
03C8  F7 77 00 00 77 77 00 00  : 5C
03D0  8F 8F 00 00 07 8F 00 00  : B4
03D8  77 77 00 00 F7 77 00 00  : 5C
03E0  EF 77 00 00 DF 8F 00 00  : D4
03E8  DF 77 00 00 DF 77 00 00  : AC
03F0  DF 77 00 00 DF 77 00 00  : AC
03F8  DF 8F 00 00 8F 6F 00 00  : 6C
---------------------------------
SUM:  C8 48 00 00 20 C8 00 00  E93C

0400  77 57 00 00 77 57 00 00  : 9C
0408  77 57 00 00 77 57 00 00  : 9C
0410  87 57 00 00 F7 57 00 00  : 2C
0418  F7 57 00 00 77 57 00 00  : 1C
0420  8F 6F 00 00 C7 8F 00 00  : 54
0428  EF 77 00 00 EF 77 00 00  : CC
0430  EF 77 00 00 EF 77 00 00  : CC
0438  EF 77 00 00 6F 77 00 00  : 4C
0440  6F 17 00 00 6F 6F 00 00  : 64
0448  9F 97 00 00 77 FF 00 00  : AC
0450  77 FF 00 00 6F FF 00 00  : E4
0458  6F FF 00 00 5F FF 00 00  : CC
0460  5F FF 00 00 2F FF 00 00  : 8C
0468  2F FF 00 00 77 FF 00 00  : A4
0470  77 FF 05 FF 00 F1 0C FF  : 76
0478  00 1F 0B FF 00 11 00 1F  : 59
---------------------------------
SUM:  C1 F8 10 FE CA BC 0C 1E  3411

0480  0A FF 01 F1 0A FF 02 F1  : F7
0488  09 FF 00 F1 02 11 09 FF  : 14
0490  00 1F 00 11 01 1F 08 FF  : 57
0498  03 11 00 1F 07 FF 00 11  : 4A
04A0  00 F1 01 11 00 F1 00 1F  : 13
04A8  06 FF 00 F1 04 11 06 FF  : 10
04B0  00 F1 00 1F 00 F1 01 11  : 13
04B8  00 FF 00 11 05 FF 00 F1  : 05
04C0  00 11 00 F1 02 11 00 F1  : 06
04C8  00 11 05 FF 06 11 00 1F  : 4B
04D0  04 FF 00 11 00 FF 03 11  : 27
04D8  00 1F 00 F1 00 1F 03 FF  : 31
04E0  00 11 00 1F 04 11 00 1F  : 64
04E8  00 11 00 1F 02 FF 09 11  : 4B
04F0  00 1F 01 FF 00 F1 00 1F  : 2F
04F8  00 F1 05 11 00 FF 00 11  : 17
---------------------------------
SUM:  20 80 0D 84 2B 60 29 A0  68E8

0500  01 FF 00 F1 00 11 00 F1  : F3
0508  01 11 00 1F 00 FF 02 11  : 43
0510  00 F1 00 11 01 FF 03 11  : 16
0518  00 1F 00 11 00 1F 03 11  : 63
0520  00 1F 00 FF 00 11 00 FF  : 2E
0528  01 11 01 1F 00 FF 00 1F  : 50
0530  01 11 00 1F 00 F1 00 1F  : 41
0538  00 F1 00 1F 01 11 01 1F  : 42
0540  00 FF 01 1F 01 11 00 1F  : 50
0548  00 11 00 F1 02 11 01 1F  : 35
```

```
0550  00 F1 00 11 01 1F 04 11  : 37
0558  00 1F 01 11 00 F1 01 FF  : 22
0560  00 1F 01 F1 00 11 00 1F  : 41
0568  01 11 00 FF 00 F1 00 11  : 13
0570  01 1F 00 F1 00 11 00 FF  : 21
0578  00 1F 01 11 00 FF 00 F1  : 21
---------------------------------
SUM:  06 E0 05 B2 06 84 0F EE  077A

0580  00 11 00 FF 01 11 00 F1  : 13
0588  00 FF 01 1F 01 F1 00 11  : 22
0590  00 1F 00 F1 03 11 01 1F  : 44
0598  00 FF 00 11 00 FF 00 1F  : 2E
05A0  07 11 01 1F 00 F1 00 FF  : 28
05A8  00 1F 04 11 00 1F 02 11  : 66
05B0  01 1F 00 FF 00 1F 02 11  : 51
05B8  00 1F 01 11 00 FF 02 11  : 43
05C0  00 1F 00 11 00 1F 02 11  : 62
05C8  00 1F 00 F1 00 11 00 FF  : 20
05D0  03 11 00 1F 00 FF 03 11  : 46
05D8  00 1F 00 F1 0C 11 00 1F  : 4C
05E0  00 11 00 1F 07 11 00 1F  : 67
05E8  00 11 00 1F 00 F1 00 1F  : 40
05F0  00 F1 07 11 00 FF 00 11  : 19
05F8  00 FF 01 F1 00 1F 05 11  : 26
---------------------------------
SUM:  0B 1C 0F B2 18 A0 11 12  EB9C

0600  00 1F 00 11 00 F1 01 FF  : 21
0608  00 11 00 1F 00 11 01 F1  : 33
0610  01 11 01 F1 00 1F 00 11  : 34
0618  00 1F 01 FF 01 11 00 F1  : 22
0620  00 11 01 1F 01 11 00 F1  : 34
0628  00 11 00 1F 02 FF 02 11  : 44
0630  02 1F 02 11 00 1F 03 FF  : 55
0638  01 11 00 FF 01 F1 00 FF  : 02
0640  00 F1 00 11 00 1F 07 FF  : 27
0648  01 1F 0A FF 00 11 00 F1  : 2B
0650  00 1F 09 FF 00 F1 00 FF  : 17
0658  00 F1 09 FF 00 F1 00 1F  : 09
0660  00 FF 00 11 09 FF 02 11  : 2B
0668  00 1F 08 FF 03 11 00 1F  : 59
0670  07 FF 00 F1 03 11 3A FF  : 44
0678  02 F1 00 11 00 F1 00 11  : 06
---------------------------------
SUM:  0E E0 29 8E 14 76 4A 40  A2EF

0680  00 F1 00 11 00 F1 00 11  : 04
0688  00 F1 00 11 01 FF 02 1F  : 23
0690  00 FF 07 1F 01 FF 00 1F  : 44
0698  00 F1 00 11 00 F1 00 11  : 04
06A0  05 F1 01 FF 0B 1F 00 FF  : 1F
06A8  02 F1 00 11 00 F1 00 11  : 06
06B0  00 F1 00 11 00 F1 00 11  : 04
06B8  00 F1 00 11 00 FF 00 03  : 04
06C0  15 33 00 03 8F 33 0F DD  : F9
06C8  00 D3 05 33 00 3D 0E 33  : 89
06D0  00 3D 06 33 00 D3 0E 33  : 8A
06D8  00 D3 04 33 00 3D 00 DD  : 24
06E0  0E 33 00 3D 00 DD 04 33  : 92
06E8  00 D3 10 33 00 D3 03 33  : 1F
06F0  00 3D 10 33 00 3D 04 33  : F4
06F8  00 D3 10 33 00 D3 03 33  : 1F
---------------------------------
SUM:  2A C2 47 F6 9C 20 3B 70  86D0

0700  00 3D 10 33 00 3D 04 33  : F4
0708  00 D3 10 33 00 D3 03 33  : 1F
0710  00 3D 10 33 00 3D 04 33  : F4
0718  00 D3 10 33 00 D3 03 33  : 1F
0720  00 3D 10 33 00 3D 04 33  : F4
0728  00 D3 10 33 00 D3 03 33  : 1F
0730  00 3D 10 33 00 3D 04 33  : F4
0738  00 D3 10 33 00 D3 03 33  : 1F
0740  00 3D 10 33 00 3D 04 33  : F4
0748  00 D3 10 33 00 D3 03 33  : 1F
0750  00 3D 10 33 00 3D 04 33  : F4
0758  00 D3 10 33 00 D3 03 33  : 1F
0760  00 3D 10 33 00 3D 04 33  : F4
0768  00 D3 10 33 00 D3 03 33  : 1F
0770  00 3D 10 33 00 3D 04 33  : F4
0778  00 D3 07 33 00 D3 07 33  : 1A
---------------------------------
SUM:  00 80 F7 30 00 80 3C 30  4A3A

0780  00 D3 03 33 00 3D 07 33  : 80
0788  00 3D 07 33 00 3D 04 33  : EB
0790  00 D3 07 33 00 D3 07 33  : 1A
0798  00 D3 03 33 00 3D 07 33  : 80
07A0  00 3D 07 33 00 3D 04 33  : EB
07A8  00 D3 07 33 00 D3 07 33  : 1A
07B0  00 D3 03 33 00 3D 07 33  : 80
07B8  00 3D 07 33 00 3D 04 33  : EB
07C0  00 D3 06 33 00 3D 00 DD  : 26
07C8  07 33 00 D3 03 33 00 3D  : 80
07D0  07 33 00 DD 00 D3 06 33  : 23
07D8  00 3D 04 33 00 D3 04 33  : 7E
07E0  00 D3 00 33 00 3D 00 DD  : 20
07E8  01 33 00 D3 04 33 00 D3  : 11
07F0  03 33 00 3D 04 33 00 3D  : E7
```

▶表紙もつるつるに戻ったし(ちょっと違うけど),K. Y. 氏も復活したし，あとは清水和人氏の復活だな。でも，本を開けたときの「バリバリ」が復活したらやだなー。
富田　祐樹 (17) X68000ACE-HD, X1turbo　東京都

```
07F8  01 33 01 D3 00 33 00 3D  : 78
-----------------------------------
SUM:  13 B8 37 C4 0B 00 39 42  A026

0800  04 33 00 3D 04 33 00 D3  : 7E
0808  04 33 00 3D 00 33 01 3D  : E5
0810  00 33 00 3D 05 33 00 D3  : 7B
0818  03 33 00 3D 05 33 00 3D  : E8
0820  00 33 01 D3 00 3D 05 33  : 7C
0828  00 3D 04 33 00 D3 05 33  : 7F
0830  00 D3 01 3D 00 33 00 D3  : 17
0838  05 33 00 D3 03 33 00 3D  : 7E
0840  06 33 03 D3 05 33 00 3D  : 84
0848  04 33 00 D3 06 33 00 DD  : 20
0850  00 3D 00 D3 06 33 00 D3  : 1C
0858  03 33 00 3D 06 33 00 3D  : E9
0860  00 D3 00 DD 06 33 00 3D  : 26
0868  04 33 00 D3 04 33 00 3D  : 7E
0870  01 DD 00 3D 01 DD 05 33  : 31
0878  00 D3 03 33 00 3D 04 33  : 7D
-----------------------------------
SUM:  22 CE 0C E0 33 8E 14 A0  998C

0880  02 DD 00 33 00 3D 01 DD  : 2D
0888  00 D3 03 33 00 3D 04 33  : 7D
0890  00 D3 02 33 02 DD 03 33  : 1D
0898  00 3D 01 DD 00 D3 02 33  : 23
08A0  00 D3 03 33 00 3D 04 33  : 7D
08A8  02 DD 00 33 00 3D 01 DD  : 2D
08B0  00 D3 03 33 00 3D 04 33  : 7D
08B8  00 D3 04 33 00 3D 01 DD  : 25
08C0  00 3D 01 DD 05 33 00 D3  : 26
08C8  03 33 00 3D 06 33 00 3D  : E9
08D0  00 D3 00 DD 06 33 00 3D  : 26
08D8  04 33 00 D3 06 33 00 DD  : 20
08E0  00 3D 00 D3 06 33 00 D3  : 1C
08E8  03 33 00 3D 06 33 03 D3  : 82
08F0  05 33 00 3D 04 33 00 D3  : 7F
08F8  05 33 00 D3 01 3D 00 33  : 7C
-----------------------------------
SUM:  18 62 11 2C 2A C0 17 6C  4D41

0900  00 D3 05 33 00 D3 03 33  : 14
0908  00 3D 05 33 00 3D 00 33  : E5
0910  01 D3 00 3D 05 33 00 3D  : 86
0918  04 33 00 D3 04 33 00 3D  : 7E
0920  00 33 01 3D 00 33 00 3D  : E1
0928  05 33 00 D3 03 33 00 3D  : 7E
0930  04 33 00 3D 01 33 01 D3  : 7C
0938  00 33 00 3D 04 33 00 3D  : E4
0940  04 33 00 D3 04 33 00 D3  : 14
0948  00 33 00 3D 00 DD 01 33  : 81
0950  00 D3 04 33 00 D3 03 33  : 13
0958  00 3D 07 33 00 DD 00 D3  : 27
0960  06 33 00 3D 04 33 00 D3  : 80
0968  06 33 00 3D 00 DD 07 33  : 8D
0970  00 D3 03 33 00 3D 07 33  : 80
0978  00 3D 07 33 00 3D 04 33  : EB
-----------------------------------
SUM:  1E CE 20 56 19 8C 1A E2  CEF3

0980  00 D3 07 33 00 D3 07 33  : 1A
0988  00 D3 03 33 00 3D 07 33  : 80
0990  00 3D 07 33 00 3D 04 33  : EB
0998  00 D3 07 33 00 D3 07 33  : 1A
09A0  00 D3 03 33 00 3D 07 33  : 80
09A8  00 3D 07 33 00 3D 04 33  : EB
09B0  00 D3 07 33 00 D3 07 33  : 1A
09B8  00 D3 03 33 00 3D 10 33  : 89
09C0  00 3D 04 33 00 D3 10 33  : 8A
09C8  00 D3 03 33 00 3D 10 33  : 89
09D0  00 3D 04 33 00 D3 10 33  : 8A
09D8  00 D3 03 33 00 3D 10 33  : 89
09E0  00 3D 04 33 00 D3 10 33  : 8A
09E8  00 D3 03 33 00 3D 10 33  : 89
09F0  00 3D 04 33 00 D3 10 33  : 8A
09F8  00 D3 03 33 00 3D 10 33  : 89
-----------------------------------
SUM:  00 AC 48 30 00 EA BB 30  2B9E

0A00  00 3D 04 33 00 D3 10 33  : 8A
0A08  00 D3 03 33 00 3D 10 33  : 89
0A10  00 3D 04 33 00 D3 10 33  : 8A
0A18  00 D3 03 33 00 3D 10 33  : 89
0A20  00 3D 04 33 00 D3 10 33  : 8A
0A28  00 D3 03 33 00 3D 10 33  : 89
0A30  00 3D 04 33 00 D3 10 33  : 8A
0A38  00 D3 03 33 00 3D 10 33  : 89
0A40  00 3D 04 33 00 D3 10 33  : 8A
0A48  00 D3 03 33 00 3D 00 DD  : 23
0A50  0E 33 00 3D 00 DD 05 33  : 93
0A58  00 D3 0E 33 00 D3 05 33  : 1F
0A60  00 3D 0E 33 00 3D 06 33  : F4
0A68  0F DD 00 D3 8F 33 00 03  : 84
0A70  15 33 00 00 00 0F 15 FF  : 6B
0A78  00 0F 2F FF 00 F1 16 FF  : 43
-----------------------------------
SUM:  32 B2 6E 73 8F 70 CB 42  7AA5

0A80  00 1F 14 FF 01 F1 15 FF  : 38
```

```
0A88  01 1F 05 FF 00 F1 03 11  : 29
0A90  0A FF 00 1F 05 FF 00 11  : 3D
0A98  03 FF 00 F1 00 1F 0F FF  : 20
0AA0  00 1F 05 FF 00 1F 08 FF  : 49
0AA8  00 1F 02 FF 00 1F 00 FF  : 3E
0AB0  00 1F 06 FF 00 1F 00 FF  : 42
0AB8  00 1F 05 FF 01 1F 02 FF  : 44
0AC0  00 11 08 FF 00 F1 00 1F  : 28
0AC8  05 FF 01 F1 03 FF 00 F1  : E9
0AD0  07 FF 00 F1 07 FF 01 1F  : 1D
0AD8  00 FF 00 F1 00 11 0A FF  : 0A
0AE0  00 F1 00 1F 05 FF 00 1F  : 33
0AE8  02 FF 00 F1 00 1F 08 FF  : 18
0AF0  00 11 1D FF 01 F1 02 FF  : 20
0AF8  00 F1 09 FF 00 F1 06 FF  : EF
-----------------------------------
SUM:  1C B8 5A EA 17 7C 4C 66  4193

0B00  01 1F 02 FF 00 1F 01 FF  : 40
0B08  00 F1 02 FF 00 F1 02 FF  : E4
0B10  00 1F 05 FF 00 F1 00 1F  : 33
0B18  02 FF 00 1F 02 FF 00 1F  : 40
0B20  00 FF 00 1F 00 FF 00 1F  : 3C
0B28  02 FF 00 1F 05 FF 01 1F  : 44
0B30  01 FF 01 F1 02 FF 03 1F  : 15
0B38  01 FF 01 F1 05 FF 01 F1  : E8
0B40  02 FF 00 1F 00 F1 00 FF  : 10
0B48  00 1F 03 F1 00 FF 00 1F  : 31
0B50  00 F1 00 FF 00 1F 09 FF  : 17
0B58  00 F1 01 FF 00 1F 00 FF  : 0F
0B60  00 F1 00 FF 00 F1 01 FF  : E1
0B68  00 1F 00 FF 00 F1 05 FF  : 13
0B70  00 F1 00 11 02 FF 01 1F  : 23
0B78  00 F1 05 FF 00 F1 00 FF  : E5
-----------------------------------
SUM:  09 1C 14 58 10 FC 18 C2  6F0C

0B80  01 1F 05 FF 00 1F 02 FF  : 44
0B88  00 F1 00 FF 00 F1 06 FF  : E6
0B90  00 F1 00 FF 00 F1 05 FF  : E5
0B98  00 F1 00 11 02 FF 00 1F  : 22
0BA0  00 FF 00 1F 06 FF 00 1F  : 42
0BA8  00 FF 00 1F 05 FF 00 1F  : 41
0BB0  02 FF 00 F1 00 FF 00 1F  : 10
0BB8  07 FF 00 1F 00 F1 05 FF  : 1A
0BC0  00 F1 00 11 02 FF 00 1F  : 22
0BC8  00 F1 07 FF 00 F1 00 FF  : E7
0BD0  00 1F 09 FF 01 F1 00 FF  : 18
0BD8  00 F1 00 11 00 1F 01 FF  : 21
0BE0  01 11 00 FF 01 F1 05 FF  : 07
0BE8  00 F1 00 11 02 FF 00 1F  : 22
0BF0  00 FF 00 F1 01 FF 00 1F  : 0F
0BF8  00 FF 00 1F 00 FF 00 F1  : 0E
-----------------------------------
SUM:  0B E0 15 9C 14 DC 18 C2  76D5

0C00  01 FF 00 1F 05 FF 01 1F  : 43
0C08  00 FF 00 F1 01 1F 00 FF  : 0F
0C10  00 1F 00 FF 00 F1 00 FF  : 0E
0C18  00 F1 01 FF 00 1F 00 FF  : 0F
0C20  00 1F 00 11 04 FF 00 F1  : 24
0C28  00 1F 00 FF 00 F1 00 FF  : 0E
0C30  00 11 09 FF 00 F1 00 1F  : 29
0C38  00 F1 04 FF 01 1F 00 FF  : 13
0C40  00 1F 00 F1 00 1F 09 FF  : 37
0C48  00 11 00 FF 00 1F 03 FF  : 31
0C50  01 F1 00 FF 02 F1 02 FF  : E5
0C58  00 1F 02 FF 00 1F 01 FF  : 3F
0C60  02 F1 07 FF 02 1F 09 FF  : 22
0C68  02 1F 06 FF 02 F1 09 FF  : 21
0C70  02 F1 07 FF 02 1F 09 FF  : 22
0C78  02 1F 06 FF 00 F1 00 FF  : 16
-----------------------------------
SUM:  0A AE 2A 06 13 9C 2B 22  7DF4

0C80  00 F1 03 FF 00 F1 00 11  : F5
0C88  03 FF 00 F1 00 FF 00 F1  : E3
0C90  07 FF 00 F1 00 FF 00 1F  : 15
0C98  02 FF 00 F1 00 FF 00 F1  : E2
0CA0  03 FF 00 1F 00 F1 08 FF  : 19
0CA8  00 1F 00 F1 02 FF 00 F1  : 02
0CB0  01 FF 00 F1 02 FF 00 F1  : E3
0CB8  00 FF 00 1F 08 FF 00 11  : 36
0CC0  00 1F 00 FF 00 F1 00 FF  : 0E
0CC8  00 1F 01 FF 00 1F 00 FF  : 3D
0CD0  00 F1 00 FF 00 11 00 1F  : 20
0CD8  09 FF 00 F1 00 FF 00 F1  : E9
0CE0  00 FF 00 F1 01 FF 00 F1  : E1
0CE8  00 FF 00 F1 00 FF 00 F1  : E0
0CF0  0B FF 00 1F 02 FF 00 F1  : 1B
0CF8  00 FF 00 F1 03 FF 00 1F  : 11
-----------------------------------
SUM:  24 34 04 D2 12 F8 08 04  3A38

0D00  0A FF 00 F1 01 FF 00 1F  : 19
0D08  00 FF 00 F1 00 11 02 FF  : 02
0D10  00 1F 00 F1 0B FF 00 1F  : 39
0D18  09 FF 00 1F 0A FF 00 F1  : 21
0D20  09 FF 00 F1 0B FF 00 1F  : 22
0D28  09 FF 00 1F 0B FF 00 1F  : 50
```

```
0D30  02 FF 00 F1 00 11 03 FF  : 05
0D38  00 1F 0B FF 00 F1 08 FF  : 21
0D40  00 F1 0B FF 00 11 00 B1  : BD
0D48  08 FF 00 B1 00 1F 09 FF  : DF
0D50  00 1B 00 BB 00 B1 05 FF  : 8B
0D58  01 F1 00 FB 00 BB 00 11  : B9
0D60  08 FF 00 1B 01 BB 00 B1  : 8F
0D68  04 FF 02 F1 01 BB 00 B1  : 63
0D70  07 FF 00 1B 02 BB 00 B1  : 8F
0D78  03 11 02 1F 00 1B 01 BB  : 0C
-----------------------------------
SUM:  46 42 1A 9E 30 F6 1C F8  9FE0

0D80  00 B1 06 FF 00 1B 08 BB  : 94
0D88  00 B1 01 F1 02 BB 00 B1  : 11
0D90  05 FF 00 1B 09 BB 01 1F  : 03
0D98  00 1B 02 BB 00 1F 04 FF  : FA
0DA0  00 F1 00 11 00 1B 07 BB  : DF
0DA8  00 B1 01 F1 02 BB 00 15  : 75
0DB0  00 1F 03 FF 00 F1 00 15  : 27
0DB8  00 55 00 1B 07 BB 01 1F  : 52
0DC0  00 1B 01 BB 00 B1 00 55  : DD
0DC8  04 FF 00 15 01 55 00 1B  : 89
0DD0  06 BB 00 B1 01 FF 00 1B  : 8D
0DD8  01 BB 00 15 00 5F 02 FF  : 31
0DE0  00 F3 09 33 00 3B 00 B1  : 1B
0DE8  00 F1 00 1F 00 F1 01 BB  : BD
0DF0  00 15 00 55 02 FF 00 F3  : 5E
0DF8  0A 33 00 3B 00 1F 00 F1  : 88
-----------------------------------
SUM:  1A 4E 17 5A 18 E0 18 68  8FE7

0E00  00 FF 00 1B 00 BB 00 B1  : 86
0E08  00 55 00 5F 02 FF 00 3D  : F2
0E10  03 DD 00 D3 00 3D 03 DD  : D0
0E18  00 D3 00 B1 00 F1 00 1F  : 94
0E20  00 F1 01 BB 00 15 00 55  : 17
0E28  02 FF 00 F3 00 3D 03 33  : 67
0E30  01 D3 02 33 00 D3 00 3B  : 17
0E38  00 1F 01 FF 00 11 00 1B  : 4B
0E40  00 B1 00 55 00 5F 02 FF  : 66
0E48  00 33 00 D3 02 33 00 3D  : 78
0E50  00 33 00 D3 01 33 00 D3  : 0D
0E58  00 33 00 BB 01 11 00 15  : 15
0E60  00 55 00 11 01 55 02 FF  : BD
0E68  00 F3 00 33 00 D3 02 33  : 2E
0E70  01 3D 01 33 00 3D 00 33  : E2
0E78  00 3B 00 BB 00 15 02 55  : 62
-----------------------------------
SUM:  07 F0 05 C6 07 6E 0E A6  D401

0E80  00 51 00 55 03 FF 00 33  : DB
0E88  00 3D 03 33 00 D3 00 3D  : 83
0E90  00 33 00 3D 01 33 00 BB  : 5F
0E98  00 B1 04 55 00 5F 02 FF  : 6A
0EA0  00 F3 00 33 00 3D 03 33  : 99
0EA8  01 D3 00 33 00 D3 00 33  : 0D
0EB0  00 3B 00 BB 00 15 04 55  : 64
0EB8  01 F1 01 FF 01 33 00 D3  : F9
0EC0  02 33 00 3D 00 33 01 D3  : 79
0EC8  01 33 00 BB 00 B1 04 55  : F9
0ED0  00 5F 01 1F 00 FF 00 F3  : 71
0ED8  01 33 00 D3 02 33 02 3D  : 7B
0EE0  01 33 00 3B 00 BB 00 B1  : DB
0EE8  04 55 00 FF 00 11 01 FF  : 69
0EF0  01 33 00 3D 03 33 00 D3  : 7A
0EF8  00 3D 02 33 01 BB 00 15  : 43
-----------------------------------
SUM:  0C 54 0B CE 0B 8C 11 A8  9663

0F00  03 55 00 5F 01 1F 00 FF  : D6
0F08  00 F3 01 33 00 3D 03 33  : 9A
0F10  01 D3 01 33 00 3B 00 BB  : FE
0F18  00 B1 03 55 00 51 00 F1  : 4B
0F20  00 11 01 FF 02 33 00 D3  : 19
0F28  02 33 00 3D 03 33 01 BB  : 64
0F30  00 B1 03 55 00 1F 02 FF  : 29
0F38  00 F3 02 33 00 D3 02 33  : 30
0F40  00 3D 02 33 00 3B 01 BB  : 69
0F48  00 15 02 55 00 1F 00 F1  : 7C
0F50  00 11 01 FF 02 33 00 3D  : 83
0F58  07 33 02 BB 00 15 01 55  : 62
0F60  00 1F 01 FF 00 1F 00 FF  : 3D
0F68  00 F3 02 33 00 3D 00 33  : 98
0F70  00 3F 0C FF 00 F1 00 11  : 4C
0F78  01 FF 02 33 01 D3 00 33  : 3C
-----------------------------------
SUM:  0E 9A 23 84 09 02 0A 52  987A

0F80  00 F1 0B 11 00 1F 00 FF  : 2B
0F88  00 1F 00 FF 00 F3 01 33  : 45
0F90  00 3D 00 33 00 D3 00 3F  : 82
0F98  00 11 00 FF 00 11 00 1F  : 40
0FA0  00 F1 00 11 00 FF 00 11  : 12
0FA8  00 1F 00 F1 00 11 00 FF  : 20
0FB0  00 11 00 F1 00 11 01 FF  : 13
0FB8  01 33 02 3D 00 33 01 F1  : 98
0FC0  01 1F 00 11 01 F1 01 1F  : 43
0FC8  00 11 01 F1 01 1F 02 FF  : 24
0FD0  00 F3 01 33 01 D3 00 3D  : 38
```

▶なんと「ちゃだワ」も，もう５周年ではないですか。めでたいめでたい。というわけで，
ちゃだワ常連を目指す（全然自慢にもならんな）私としては，今年も投稿だ〜。

中村　祐一（19）MZ-2500　東京都

```
0FD8  00 3F 00 1F 01 11 00 F1  : 61
0FE0  01 1F 00 11 01 F1 01 1F  : 43
0FE8  00 11 02 F1 01 FF 01 33  : 38
0FF0  00 D3 00 33 01 D3 01 F1  : CC
0FF8  00 11 00 1F 00 11 01 F1  : 33
-----------------------------------
SUM:  03 28 11 1A 07 12 0A 10  2899

1000  01 1F 00 11 01 F1 03 1F  : 45
1008  00 FF 00 F3 00 33 00 3D  : 62
1010  00 33 00 3D 00 33 00 3F  : E2
1018  00 1F 00 FF 00 11 00 1F  : 4E
1020  00 F1 00 1F 00 11 01 F1  : 13
1028  01 1F 00 11 00 F1 00 FF  : 21
1030  00 11 01 FF 00 33 00 3D  : 81
1038  01 33 00 3D 00 33 01 F1  : 96
1040  01 1F 00 11 01 F1 01 1F  : 43
1048  00 11 01 F1 03 1F 00 FF  : 24
1050  00 F3 00 33 00 D3 01 33  : 2D
1058  00 D3 00 3F 00 1F 00 11  : 42
1060  01 F1 01 1F 00 11 01 F1  : 15
1068  01 1F 00 11 02 F1 01 FF  : 24
1070  00 33 00 D3 02 33 00 D3  : 0E
1078  01 F1 01 1F 00 11 01 F1  : 15
-----------------------------------
SUM:  07 EE 04 42 09 18 0A EE  CB72

1080  01 1F 00 11 01 F1 01 1F  : 43
1088  02 FF 00 F3 04 DD 00 3F  : 14
1090  00 1F 00 11 01 F1 01 1F  : 42
1098  00 11 01 F1 01 1F 00 11  : 34
10A0  00 F1 00 FF 00 1F 01 FF  : 0F
10A8  05 33 00 F1 00 1F 00 F1  : 39
10B0  00 11 00 FF 00 11 00 1F  : 40
10B8  00 F1 00 11 00 FF 00 11  : 12
10C0  00 1F 00 F1 02 1F 01 FF  : 31
10C8  04 33 00 3F 0C 11 01 F1  : 85
10D0  15 FF 01 1F 15 FF 00 1F  : 67
10D8  2D FF 00 1F 15 FF 01 1F  : 7F
10E0  14 FF 01 F1 15 FF 00 1F  : 38
10E8  15 FF 00 F1 30 FF 00 0F  : 43
10F0  15 FF FF FF FC 44 44 44  : DA
10F8  4F FF CF FF C4 10 10 4F  : 4F
-----------------------------------
SUM:  DB C0 D1 54 44 AC 5A 9D  51F2

1100  FF F3 FF F1 55 55 53 FF  : DE
1108  F0 3F FC 00 00 00 FF FC  : 26
1110  0F FF C0 00 00 F3 FC 00  : BD
1118  FF F2 20 FF 38 BF 00 3F  : 46
1120  FC 88 3C 0C 4F 00 03 FC  : 1A
1128  88 8C F3 8B C0 00 FF 22  : 73
1130  23 C0 F7 F0 00 3F 02 20  : 2B
1138  FC 38 BF 33 3F C8 00 FF  : 2C
1140  F2 2F F0 3F C2 22 3F FC  : 6F
1148  DF F0 03 F2 08 8F FC 22  : 79
1150  FF FF F2 08 83 FC 31 3F  : E7
1158  FF F0 22 23 3F 3E 2D FF  : DD
1160  F0 40 80 F3 CF CF 7F C0  : 80
1168  34 4C CF 03 C3 D7 00 0D  : F9
1170  04 CC CF F3 3D 37 00 D0  : D6
1178  40 0C 33 FD 5D C3 34 45  : 15
-----------------------------------
SUM:  D7 A1 18 EC 93 99 9E B5  CA3F

1180  57 00 F3 DC 7C CD 04 01  : 74
1188  CC C3 D7 17 00 D1 54 70  : 12
1190  31 3D C1 F0 34 55 1C CC  : 90
1198  11 72 5C CD 1F 1C CC 01  : B4
11A0  DC 07 33 45 C7 03 10 17  : 4C
11A8  29 C0 D1 71 CC C9 0D C0  : 8D
11B0  70 0D 1C 73 30 91 72 97  : D6
11B8  33 47 1C 0C 89 DC 05 CC  : D8
11C0  D1 C7 33 08 17 29 70 34  : B7
11C8  71 CC C8 0D C0 5C 0D 14  : 4F
11D0  70 30 89 72 B7 F3 F5 DD  : 17
11D8  43 C8 DC 03 C3 C3 C3 55  : 88
11E0  3C 17 2B 0F 0F 0F 0D 53  : 0B
11E8  FD C0 BE BE BE BF D5 3D  : C8
11F0  72 AA 8A 8A 8A CF 53 DC  : B8
11F8  00 00 00 00 00 D4 D7 2B  : D6
-----------------------------------
SUM:  AD 99 F6 C6 C3 F4 15 89  5A46

1200  FF FF FF FC 0D 4D C0 55  : 68
1208  55 55 55 03 51 72 80 00  : 45
1210  00 00 0A 35 DC 00 00 00  : 1B
1218  00 00 0D D7 28 03 33 33  : 75
1220  00 A3 5F F3 33 3C 18 A3  : 1F
1228  FF FF FC 3C F3 C6 88 FC  : 73
1230  FF FF 3C CF 31 8A 3F 3F  : 42
1238  FF CC FC F0 68 3F 03 FF  : 60
1240  FC FF CF 18 8F C0 FF FF  : 2F
1248  03 C0 C6 A3 C0 0F FF 4F  : 49
1250  3F C0 48 F0 03 C0 D0 0C  : D6
1258  3C 1A 30 00 32 84 F3 FF  : 2E
1260  04 8C 00 0C A1 3C FF C1  : 39
1268  83 00 03 28 43 0F CC 1A  : E6
1270  33 33 F0 D0 FF CF 04 8F  : 87
1278  03 FE B4 3C F3 C1 8F 00  : 34
-----------------------------------
SUM:  88 17 B2 E4 7B 7B 74 28  A280

1280  3F 0D 03 F3 F0 0F FF FF  : 3F
1288  28 40 FC FF C0 FF F2 0A  : 1E
1290  10 00 F0 00 0F FE 22 8D  : BC
1298  08 80 88 80 FF 20 A3 0A  : 5C
12A0  22 A0 00 0F 33 28 08 AA  : DE
12A8  00 83 08 F0 0A 02 00 00  : 87
12B0  80 08 00 F2 8C A8 A0 80  : CE
12B8  C2 03 CF A0 38 00 80 C0  : AC
12C0  80 F0 28 82 8A 83 00 A3  : CA
12C8  3F 8A 2C 80 80 00 88 03  : 80
12D0  E6 82 E8 83 C0 96 3F 28  : 90
12D8  A1 82 20 02 95 80 00 E8  : 42
12E0  48 A3 C2 54 63 FC CA 06  : 30
12E8  20 0A 51 58 03 F2 85 20  : 6D
12F0  C9 45 42 3C FC AC 5A 89  : 17
12F8  15 00 80 FF 00 C6 09 14  : 77
-----------------------------------
SUM:  6F 6B 7F 71 80 F7 57 03  200A

1300  0F 08 FC CA 0D 65 50 0C  : AB
1308  C2 3C 30 03 66 50 3C 30  : 53
1310  8C 0C A0 09 80 F0 0C 23  : E0
1318  00 28 0E 60 FF FF 08 F3  : 8F
1320  CA 00 20 00 00 02 0C FA  : F2
1328  AA AA AA AA AA A3 FC A0  : 91
1330  00 00 00 00 08 FF 2A AA  : DB
1338  AA AA AA AA 3F CA 00 00  : B1
1340  00 00 00 8F F2 80 00 00  : 01
1348  00 00 20 CC A0 03 03 03  : 95
1350  03 02 AA FF F2 AA AA AA  : 9E
1358  AA 3F FF CF 28 88 88 8A  : 79
1360  3F FF F3 F2 8A 8A 8A 3F  : 00
1368  FF F0 3C 00 00 00 0F FF  : 39
1370  FC 0F CA AA AA 8F FF FC  : B3
1378  00 F1 55 55 53 FF 0F 00  : FC
-----------------------------------
SUM:  62 FC 65 A4 16 DF AE 07  A68A

1380  3F 00 00 03 FF C3 00 03  : 07
1388  CF 3F FC FF CF C0 00 F3  : 8B
1390  C0 C0 3F F3 F0 00 3C F0  : CE
1398  30 03 FC FF 33 3F 3C FC  : D8
13A0  F0 FF 3B F0 3C 0F 3C 3C  : DD
13A8  3F CB F0 03 33 CC F3 0F  : FE
13B0  F3 FF FF CC F3 FF CC FC  : 77
13B8  FF FF 03 3C FF F3 3F 2F  : 9D
13C0  FC 20 FF 3F 3C 3F CF F0  : 94
13C8  02 0F 03 03 F3 F3 B0 40  : ED
13D0  B3 C3 CC 3C 00 B0 40 0B  : 79
13D8  3F 0C C0 08 3E 12 02 0F  : 74
13E0  C0 0E 08 0E 04 00 B0 00  : 98
13E8  06 02 02 C1 00 23 96 25  : A9
13F0  8C 80 F2 40 88 25 89 6C  : E0
13F8  80 38 90 00 89 62 58 23  : AE
-----------------------------------
SUM:  E1 90 7E 84 D4 2D 9A 56  7ED4

1400  CB 04 80 2C 96 96 38 03  : E2
1408  C1 00 08 E5 A5 B2 30 E0  : 15
1410  40 82 09 65 80 80 2E 04  : 62
1418  00 B2 59 63 A0 CF 89 00  : 66
1420  23 96 5B 23 33 A2 48 88  : DC
1428  25 96 08 0C B8 10 02 C9  : 62
1430  65 8E 0D 3E 01 00 8E 59  : 26
1438  6C 83 4E 60 42 20 96 68  : FD
1440  23 40 14 24 0B 2F F8 E0  : AD
1448  D0 05 82 42 38 02 C8 35  : D0
1450  3D 50 04 82 00 82 3D 4C  : 1E
1458  06 02 6C 8A 23 83 53 C0  : B7
1460  A0 28 E0 0B 20 D4 C0 1A  : 81
1468  02 08 82 08 35 3C 06 A0  : AB
1470  BE 22 02 CD 4C 51 AA 20  : 16
1478  88 8E 02 54 55 6A A8 EA  : BD
-----------------------------------
SUM:  03 EC 14 4C E5 6A F5 DE  0ED5

1480  AC A2 95 5F FF 1D DD DD  : 18
1488  D3 FF FC F3 F1 F7 F7 D3  : 73
1490  FF FF 4F 3C 55 55 54 F5  : 7C
1498  D7 D3 23 C0 00 00 F5 75  : F7
14A0  74 28 F0 00 00 3D 55 5D  : 7B
14A8  2A 3C 88 8F CF 55 57 4A  : 42
14B0  8F 22 23 03 D5 55 D0 A3  : 74
14B8  C2 20 F0 F5 55 74 C8 C8  : 20
14C0  88 3F 3F 55 7D 3C F0 88  : 8C
14C8  8F CF D5 5F 4C C0 22 23  : E3
14D0  CC FD 5F D3 C2 22 23 CC  : CE
14D8  FF 57 F4 F0 88 88 FF 3F  : 88
14E0  F7 FD 3C 88 88 FF 0F FD  : 4B
14E8  FF 4F C2 20 CF F3 23 FF  : 14
14F0  D3 FC 20 CC 33 E2 3F F4  : 03
14F8  F0 80 CC C0 F2 23 FD 00  : 0E
-----------------------------------
SUM:  DF 43 DF 80 CD 61 03 D2  C141

1500  08 00 02 3F 22 FF 00 80  : EA
1508  88 88 83 F2 0F C0 20 08  : 7C
1510  88 80 3F 3C F0 08 20 00  : 9B
1518  08 30 CF 3C 3E 09 55 56  : 35
1520  0C 03 CF 00 82 55 55 83  : 8D
1528  02 03 CF 90 90 01 60 C0  : 15
1530  A8 30 24 25 3C 58 30 20  : 05
1538  43 E6 02 4C 18 0C 0A 91  : 36
1540  09 80 90 16 03 02 04 7E  : B6
1548  68 25 16 00 E0 A9 10 9A  : D6
1550  02 45 80 D8 20 47 E6 90  : 7C
1558  95 80 C5 8A 91 09 A4 05  : A7
1560  60 20 62 04 7A 69 82 60  : AB
1568  3A 18 A9 12 5A 60 98 0E  : 6D
1570  06 20 93 96 98 26 0E 21  : 3C
1578  8A A4 25 A6 0A 83 A8 61  : 8F
-----------------------------------
SUM:  4B BA 05 74 CF F7 F2 6F  604B

1580  81 29 69 80 00 02 18 6A  : 17
1588  42 5A 7F FF FF 86 18 12  : C9
1590  96 93 33 33 21 86 A5 25  : 00
1598  66 FF FF F8 61 82 69 59  : 01
15A0  80 00 02 18 6A 90 56 69  : 53
15A8  14 51 86 1A 8A A5 A6 51  : 2B
15B0  45 16 69 6A FF FF CC CF  : C7
15B8  04 8A 8F FF FF F0 CC 31  : 08
15C0  22 A3 FF 5D 7C CF 30 48  : E4
15C8  23 FF 57 57 CF F3 12 28  : CC
15D0  FF D5 55 F3 FF 04 8A 3F  : E8
15D8  F5 55 70 0F 01 22 0F FD  : F8
15E0  55 5C 30 F0 4A 23 FF 55  : 92
15E8  57 4F 3F C4 88 FF F5 57  : 7C
15F0  13 CF F1 22 3F FD 55 C4  : 4A
15F8  F0 FC 12 0F FF D5 C5 3F  : E5
-----------------------------------
SUM:  84 48 27 E0 CE 90 BB 0F  9BD9

1600  FF 04 88 FF F5 71 4F FF  : 3E
1608  C0 48 23 FF 7C 53 C3 C0  : 7C
1610  00 E6 FF DF 16 3F C3 00  : DC
1618  F2 3F FF 16 8F 0F 00 0F  : F3
1620  3F FF C5 A8 3C 3C 03 30  : 56
1628  FF F1 60 F0 F0 00 30 FF  : 5F
1630  FC 40 C3 C0 20 0F 3F FF  : 2C
1638  08 80 00 8A C0 3F FC 30  : 3D
1640  A2 22 28 0F F3 F0 03 02  : E3
1648  22 22 03 30 F0 00 FC 02  : 65
1650  A0 3C CF 30 0A 33 FC 20  : 34
1658  F0 30 30 08 0C 03 2A 31  : C2
1660  4C 55 02 23 14 C2 0C 53  : FB
1668  15 40 A0 C5 3C 8F 1C C5  : 66
1670  50 A0 00 D3 23 17 31 54  : 82
1678  88 0C 34 C8 C7 0C 55 22  : DA
-----------------------------------
SUM:  80 12 91 CF 55 36 16 0F  D6FA

1680  0F 0D 32 31 C3 15 48 8F  : 2E
1688  0C D0 01 70 C5 52 23 33  : BA
1690  34 CC 73 31 54 A0 CC CD  : 31
1698  33 1C CC 56 88 33 0C D3  : 0B
16A0  1C C3 16 A2 0C C3 34 C7  : 61
16A8  30 C6 A8 83 30 CD 31 CC  : 1B
16B0  31 A2 80 CA AA AA AA 8C  : A7
16B8  A0 80 31 80 00 02 93 20  : 86
16C0  20 0C 19 55 55 90 C0 00  : 3F
16C8  03 02 55 55 60 30 00 08  : 47
16D0  C0 65 75 59 0C 00 02 30  : 31
16D8  09 77 56 03 00 80 0C 09  : 6E
16E0  77 75 80 C0 2F FF CA 22  : 46
16E8  22 22 8F FF FF FC A2 A2  : 11
16F0  A2 8F FF FF FF 08 A8 A8  : 86
16F8  83 FF FD 75 F0 80 80 83  : 67
-----------------------------------
SUM:  49 7F 25 D0 28 39 47 D1  5BED

1700  FF FF 5D 7C 8A 8A 88 FF  : 72
1708  FF 55 57 55 55 55 4F FF  : F8
1710  D5 55 C0 00 00 0C F0 F5  : DB
1718  55 73 FF C3 30 C2 3D 55  : 0E
1720  5C 0F 00 CC CC 8F 55 57  : 3E
1728  3C FC CC C0 8F F5 57 C3  : 62
1730  30 33 30 80 FD 55 F0 CC  : 21
1738  C3 30 83 3F D5 F0 F3 F0  : 5D
1740  CC 0C CF F5 7F 3F 3C CC  : 62
1748  CF 03 FF 7F 0F FF 33 33  : C4
1750  CA 3F DF F3 C3 F3 33 C2  : 86
1758  0F FF F0 FF F0 CC F2 23  : CE
1760  FF FC CC C3 0C CC 20 FF  : 81
1768  FC CC CC 03 30 22 3F FF  : 27
1770  CC CC 33 33 22 2F FF 03  : 51
1778  C0 50 00 25 57 FC 3F 01  : C8
-----------------------------------
SUM:  AE BB 5A 63 32 8C C4 04  DA18

1780  0A 8F 0A 55 FC 8F C5 56  : 9E
1788  20 F3 A0 3C A3 31 56 88  : A1
1790  FC 9A 0C A0 FC 55 88 33  : 4E
1798  25 A4 A1 4F 15 A2 0F C9  : 48
17A0  6A 21 50 F1 62 13 F2 5B  : 8E
17A8  A1 54 F0 68 84 FC 96 38  : 9B
17B0  15 33 18 87 30 49 8C 01  : ED
```

▶たしか僕が小学生のころ，某雑誌KKKでスネオ君の髪型についての討論会をやっていた。スネオ君の髪型は，上下左右前後を合成し立体感をつけてみると，理論上ありえない髪型になってしまうという話だった。　荒井　信親（16）X68000ACE-HD　東京都

```
17B8   F3 CA 21 CF 12 6C 00 0C   : 37
17C0   32 21 D3 C4 9B 02 88 F0   : FF
17C8   28 74 32 26 30 02 3F C8   : 2D
17D0   75 3C 89 8D 28 8F F2 1D   : 8D
17D8   4C 22 6C 00 23 FC 1D 53   : 69
17E0   C4 9B 12 82 00 07 54 F1   : 3F
17E8   26 30 00 8C 00 00 0C 49   : 37
17F0   8C E0 23 C0 00 0F 09 6C   : D3
17F8   C8 18 F0 00 03 C2 5B 32   : 22
------------------------------------
SUM:   B7 E8 EF 74 F1 E2 60 7A   122F

1800   16 0C AA A8 C0 96 33 95   : 92
1808   8F 15 55 3C 25 8C E5 63   : 2E
1810   C7 77 4F 09 6C FF C7 5D   : 25
1818   D7 75 D3 FF FF FC 77 75   : 05
1820   DD D3 FF FF FF 1F DF F7   : A2
1828   F4 FD FF FF F1 55 55 54   : DE
1830   FF 7F 3F FC 00 00 00 3F   : F8
1838   DF C3 FF 20 83 FF CF D5   : E7
1840   F0 FF C8 88 0C 03 F5 7C   : BF
1848   3F F0 82 3F 3C F5 57 8F   : 07
1850   FC 8A 00 C0 3F 57 CB FF   : A6
1858   22 23 F3 CF D5 F0 FF C2   : 8D
1860   08 FC F3 FD F0 3F F2 20   : 35
1868   3C 3C FF 70 0F C0 22 3C   : 14
1870   F3 3F DC 03 C0 A0 8F FF   : FF
1878   CF FF 88 F2 82 23 F0 F0   : CD
------------------------------------
SUM:   45 31 F0 BE 60 91 02 40   4C2C

1880   FF C8 BC 8A 20 3F F2 8F   : ED
1888   F0 0F 28 20 43 FC A3 FC   : 25
1890   03 00 A0 45 3C 54 0F 00   : 87
1898   10 00 51 10 41 20 00 28   : FA
18A0   02 11 11 00 38 F1 0C      : 6A
18A8   08 44 41 20 0C BC A0 C2   : D7
18B0   05 45 08 A3 0F 14 3C 08   : 5C
18B8   00 20 0A C0 12 03 C2 8A   : 4B
18C0   28 C3 A0 00 41 3C 02 80   : 8A
18C8   F0 D8 01 28 13 C0 00 F0   : B4
18D0   45 88 05 06 3F FF F0 20   : 26
18D8   08 94 81 43 FF F0 04 50   : A3
18E0   00 10 58 00 01 02 00 01   : 6C
18E8   4A 05 14 C5 02 45 00 01   : 70
18F0   41 63 03 C0 40 00 14 A0   : 5B
18F8   54 4C 48 21 58 80 04 15   : FA
------------------------------------
SUM:   55 0C 17 AA 3A 6C 41 AA   B25E

1900   80 31 04 00 89 52 82 50   : 62
1908   C2 0A 15 00 00 50 16 00   : 47
1910   41 40 00 15 28 09 4C 90   : A3
1918   A1 50 00 05 04 58 14 14   : 7A
1920   00 01 52 81 25 25 0A 15   : 3D
1928   88 00 50 49 55 41 40 08   : FF
1930   95 28 11 41 50 A1 50 00   : 50
1938   05 02 4C 58 14 00 01 52   : 12
1940   80 5C D4 0A 15 0F CA A2   : 4A
1948   1C C3 33 FF FF F2 88 84   : 0E
1950   C3 30 FF FF FC A8 21 C3   : 79
1958   3C FF DF FF C8 88 4C CC   : 81
1960   3F F7 FF F2 A2 1C FC C3   : A4
1968   FD FF FC 82 13 3F 30 FD   : F9
1970   5F FF 28 87 3F F0 3F 57   : D2
1978   FF C8 21 3F FC 0F 55 7F   : 06
------------------------------------
SUM:   7B 01 41 BE 5B 95 12 AE   D6A5

1980   F2 88 40 F0 C3 F5 7F FC   : DD
1988   08 7F F3 F0 FD 5F FC A2   : 64
1990   1C 0C 02 8F DF FF 20 1F   : D6
1998   CF 30 23 F7 FF 28 87 FF   : C6
19A0   CF 08 FD FF C8 84 3F F3   : 51
19A8   C2 3F FF FC A1 CF F3 C4   : 23
19B0   0F C8 FF C1 03 FF F1 08   : 92
19B8   F8 BF FD 4F 3C 3C 52 3C   : 09
19C0   8F FD 40 F3 FF 14 8F CF   : 30
19C8   FD 5C C0 FF 31 23 F3 FD   : 5C
19D0   50 3F CF C3 00 03 FD 5C   : 7D
19D8   20 0C 0C 00 8C 0D 50 C8   : E9
19E0   88 00 88 03 0F 53 C2 22   : 59
19E8   22 22 20 3C D3 02 22 22   : B9
19F0   22 22 0F 3F 00 88 88 88   : 2A
19F8   88 0C 3C 00 88 88 88 88   : F0
------------------------------------
SUM:   CD 05 1E A4 6C B5 5A FB   53D9

1A00   8C 00 00 A0 22 22 22 0F   : A1
1A08   0A 00 A3 00 22 22 08 CA   : C3
1A10   00 A3 2A A0 00 28 32 00   : C7
1A18   A3 08 02 AA A0 0E 00 A3   : A8
1A20   12 00 96 58 02 20 23 15   : 5A
1A28   25 26 A6 02 88 03 35 49   : FC
1A30   42 65 86 05 83 38 52 54   : 93
1A38   AA 45 85 60 32 15 25 26   : 66
1A40   61 81 52 30 85 49 49 98   : 13
1A48   61 54 80 21 52 54 A8 58   : FC
1A50   54 08 08 54 95 2A 18 15   : A4
1A58   22 02 15 2A AA AA 15 48   : 14
1A60   00 85 48 00 00 85 48 00   : 9A
1A68   21 52 15 55 21 52 00 08   : 58
1A70   54 85 55 48 54 80 FF FF   : 48
1A78   CA AA AA AA 3F FF FF FC   : 01
------------------------------------
SUM:   D3 60 61 BF ED B1 8F A4   B5B3

1A80   22 AA AA 3F FF F7 FF 22   : CC
1A88   88 88 8F F3 FD FF F0 00   : 7E
1A90   A8 8F 31 FD 5F FC AA 00   : 6A
1A98   03 20 7F 57 FF 55 6A A8   : 5F
1AA0   CB 1F 55 7F C0 05 55 72   : 4A
1AA8   07 D5 5F F3 FC 05 5C B1   : 3C
1AB0   FD 5F FC C3 F0 03 20 7F   : AD
1AB8   57 FF 3F 3C CC CB 1F F7   : 7E
1AC0   FF CC 3F 33 32 07 FD FF   : 72
1AC8   CF 3F C3 33 31 FF FF F3   : 26
1AD0   FF FC CC FC 7F FF FF 0F   : 4F
1AD8   FF 0C 3F 1F FF FF CC CF   : 02
1AE0   C3 33 C7 FF FF F3 C3 C3   : 34
1AE8   30 F1 FF 3F FC 3F CC CC   : 32
1AF0   3C 7F 03 FF 3F F0 CC CF   : 87
1AF8   1F 0C FF 33 FC 33 33 C7   : 86
------------------------------------
SUM:   95 F5 AD E8 E9 78 48 58   CA11

1B00   03 3F 33 33 03 33 F1 C3   : 92
1B08   CF F3 33 00 03 0C 73 F3   : 6A
1B10   C0 00 2A A8 00 1C FC 09   : B3
1B18   96 AC CC 80 F3 3C 16 26   : F9
1B20   33 33 24 C3 C0 55 9A CC   : C8
1B28   FF F9 4C F1 54 22 3F AA   : 94
1B30   AE 14 F1 50 09 8F 80 08   : 23
1B38   A1 4D 41 42 2F 85 56 E2   : 5D
1B40   14 45 42 9B 85 F5 89 21   : 5A
1B48   45 50 A2 85 D7 6E 52 15   : 68
1B50   50 89 85 57 58 91 21 54   : 13
1B58   22 25 5F 56 E9 52 54 20   : AB
1B60   99 7D 76 20 91 15 08 22   : 7C
1B68   F5 5D B8 09 55 08 09 95   : 0E
1B70   7D 62 30 91 42 08 95 55   : D4
1B78   5B 8F 09 42 02 60 00 02   : 99
------------------------------------
SUM:   DA 79 2D 6A 0C ED 1B FD   6000

1B80   23 30 90 8C 8A AA AA 88   : D5
1B88   C3 00 83 26 00 00 26 30   : C2
1B90   30 A3 C8 BC FC F8 8C 0C   : E3
1B98   A0 F2 67 37 36 63 0C 20   : F5
1BA0   CC 89 CD CD 88 CC 27 FC   : 66
1BA8   45 11 11 44 FF FD FF C4   : 6A
1BB0   44 44 44 FF FF 5F F1 11   : 2B
1BB8   11 11 3C 0F D7 FC 10 40   : 90
1BC0   41 3F 03 F5 7C 05 55 55   : A3
1BC8   4F C0 FD 5F 30 00 00 03   : 9E
1BD0   C0 0F 0F FC FF FC 20 F0   : E5
1BD8   03 C3 FF 03 C3 22 3C 00   : E9
1BE0   F8 FF CF 3F C8 8F 00 3C   : 98
1BE8   BF F0 0C 30 83 FC FF 0F   : 78
1BF0   FC F3 FC 20 FC 0F E3 FF   : F8
1BF8   3C FF 22 3F 03 F2 FF CF   : 5F
------------------------------------
SUM:   5E 66 A7 E5 D1 D8 21 56   C741

1C00   CF C8 83 FF FC 3F C3 FF   : 16
1C08   F0 88 3F FF 8F CA 30 F0   : 2F
1C10   88 8F FF C3 FC 8F F3 C8   : 1F
1C18   83 FF F0 F0 0C F3 F0 00   : 51
1C20   3F F2 01 88 83 C2 AA 80   : 29
1C28   F0 30 42 E0 0A 22 23 CC   : 5D
1C30   0C 18 8A A8 88 88 30 8B   : 21
1C38   04 2E 52 2A AA 38 88 81   : 99
1C40   88 94 A0 00 02 E0 0F 02   : AF
1C48   E8 A0 00 03 88 23 7C 8A   : 3C
1C50   20 55 50 2E 20 D0 EE 20   : F1
1C58   55 54 38 80 34 D8 80 54   : 41
1C60   00 28 98 8D 36 E2 14 15   : 8E
1C68   62 E4 8B 4D 88 05 15 60   : 20
1C70   89 00 43 6E 21 45 63 2E   : 31
1C78   62 13 18 80 51 63 08 92   : 5B
------------------------------------
SUM:   3B 42 76 64 60 69 E8 44   11FE

1C80   24 C6 E2 14 60 CB 94 01   : A0
1C88   31 88 05 18 C2 25 08 4C   : 11
1C90   6E 21 48 30 B9 48 13 18   : 33
1C98   80 52 30 A2 50 07 1B 88   : 9E
1CA0   56 30 AE 46 21 C6 20 15   : 96
1CA8   8C 88 91 22 71 B8 85 83   : F8
1CB0   22 E4 40 1C 62 01 63 08   : 30
1CB8   89 18 87 1B 88 60 C8 B9   : AC
1CC0   04 89 C6 20 18 C2 22 41   : B0
1CC8   00 71 B8 88 32 8B 90 42   : 40
1CD0   1C 62 08 32 02 24 D2 2A   : DA
1CD8   00 91 30 30 FF FF F2 AA   : 8B
1CE0   24 73 33 3F FF FC A0 09   : AD
1CE8   13 33 07 F0 3F 2A A2 47   : 8F
1CF0   3C 31 FC 0F F2 02 44 CF   : 7F
1CF8   C0 7F 03 FC A8 91 CF FC   : 42
------------------------------------
SUM:   23 B8 54 E1 CA 47 65 B8   9B63

1D00   1F 00 3F 00 A4 4F F0 C7   : 08
1D08   C0 0F CA 24 4F F3 F1 F0   : E0
1D10   03 F0 29 10 0C 0C 1C 00   : 60
1D18   F2 89 13 CF 0F 07 F3 FC   : 62
1D20   0A 44 FF CF C1 F0 3C A2   : AB
1D28   44 FF F3 C4 7C 0F C2 91   : D8
1D30   0F F3 F1 47 FF FF 11 C3   : 0C
1D38   FF FC 51 FF FF F4 73 3F   : F0
1D40   0C 54 7F FF FC 73 F3 FF   : 3F
1D48   15 1F FF FC 10 FF 3F 31   : AE
1D50   47 C8 F0 13 03 F3 CC 54   : 28
1D58   79 B0 17 26 0F 0F 05 1C   : A5
1D60   8C 57 09 80 00 00 47 CF   : 82
1D68   57 32 66 66 66 11 F3 D3   : 92
1D70   02 09 99 99 90 4C C0 05   : DE
1D78   28 66 66 62 18 09 55 50   : 1C
------------------------------------
SUM:   1E 9D 6C F1 75 21 C4 7F   340B

1D80   81 99 82 07 8B 17 F4 0A   : 43
1D88   00 0A 00 C3 C0 5D 40 2A   : 54
1D90   A8 00 0F 34 05 F4 A0 00   : 84
1D98   22 03 CD 50 5D 0A 88 22   : 53
1DA0   01 33 41 05 D2 8A 00 81   : 57
1DA8   13 D0 10 54 20 22 21 41   : EB
1DB0   34 F1 05 08 8A 08 67 D1   : FC
1DB8   33 10 52 82 08 1A 41 4C   : C6
1DC0   31 14 22 A2 18 A7 D3 0C   : A7
1DC8   45 08 20 86 A0 44 FF 11   : E7
1DD0   40 88 81 88 85 00 04 50   : AA
1DD8   22 21 A8 20 55 55 14 00   : C9
1DE0   00 62 22 2A AA 80 00 C0   : 98
1DE8   0A 88 A2 22 28 00 CC 02   : 4C
1DF0   20 22 AA A2 00 C0 C0 A8   : B6
1DF8   00 00 0A 80 C8 0C 22 00   : 80
------------------------------------
SUM:   C8 7B E9 6F 5D CC BD 0C   EF53

1E00   00 02 2C C8 8C E2 00 0F   : 73
1E08   FF FC 88 88 88 8F FF FF   : 20
1E10   FF C2 22 22 0F FF FC 0F   : 1E
1E18   F2 2A 2A 23 FF FF 03 FF   : 69
1E20   00 00 03 FF FF C0 FF CA   : 8A
1E28   AA A8 FF CF C0 0F F5 55   : 39
1E30   55 7F CF F0 03 FC 00 00   : 92
1E38   0F FC FC 00 FF 3F FC 33   : 74
1E40   FC FF 00 3F C0 F0 33 3F   : 5C
1E48   CF FC FF F3 CF CC CF CF   : F6
1E50   FC 0F F0 03 00 C3 FC FF   : BC
1E58   03 FF 3C FC 30 FC FF FF   : 64
1E60   FF 0C CF 33 3F CF FF FF   : 19
1E68   F3 FF CC CF CF FF FF F0   : 4A
1E70   F0 F0 CC FC FF FF FC CC   : 6E
1E78   CC CC FC FF FF FC CC CC   : 26
------------------------------------
SUM:   76 DD 5B 81 AE BD B1 01   7AD0

1E80   CC CF CF FF FC 0C 3C CC   : 79
1E88   0C 0F FF F0 F0 A1 55 2C   : 1C
1E90   8C FF C2 0F 2E 15 2C 80   : 4B
1E98   CF 0F E3 08 E1 2C 84 C3   : 1D
1EA0   40 02 30 8E 2C 85 0C 50   : 0D
1EA8   FF 8F 20 EC 21 4C 14 00   : 1B
1EB0   08 C8 8C 88 50 C5 0F F2   : FA
1EB8   3E 08 88 50 C1 40 00 8C   : AB
1EC0   88 82 17 0C 50 FF 0B 20   : A7
1EC8   88 85 0C 14 00 02 38 88   : EF
1ED0   85 30 C5 0F F0 8C 88 21   : AE
1ED8   40 C1 40 00 23 20 88 53   : 5F
1EE0   0C 50 FF 08 F8 82 14 0C   : FD
1EE8   14 08 00 B2 08 85 C0 C5   : E0
1EF0   08 B5 23 88 21 40 C1 40   : CA
1EF8   81 08 E0 88 5C 8E 50 57   : 82
------------------------------------
SUM:   36 5A 01 51 39 46 A8 8D   E18A

1F00   12 08 82 14 2A 94 08 14   : 8A
1F08   2E 08 85 C0 05 20 34 48   : 1C
1F10   88 21 40 F1 4A AC 52 E0   : 02
1F18   88 5C F3 52 03 50 88 82   : 86
1F20   14 3F D4 AA 11 3F FF 85   : A5
1F28   FF 35 2A 01 43 38 21 4F   : 4A
1F30   F5 4A 01 43 33 02 5F C5   : DC
1F38   00 00 00 00 04 00 04 00   : 08
1F40   04 00 04 00 04 00 04 00   : 10
1F48   04 00 04 00 04 00 04 00   : 10
1F50   2E 00 04 00 04 00 04 00   : 3A
1F58   04 00 04 00 1A 00 C8 00   : EA
1F60   18 00 08 00 08 00 00 00   : 28
1F68   00 00 00 00 00 00 00 00   : 00
1F70   00 00 00 00 00 00 00 00   : 00
1F78   00 00 00 00 00 00 00 00   : 00
------------------------------------
SUM:   AA 4B 51 05 35 29 6D 57   DC81
```

▶私の部屋は和室です。それで，この頃思うのですが，正座もしくはあぐらをかいて使えるパソコンディスクはないですかね。　　北村　龍彦（26）X68000PRO　東京都

リスト3 MAKE.Bas

```
10 int ST,I,J,X,Y,CT,NP
20 int r,r_ST,r_NM,r_EC,r_AN,r_CL,w,w_PR
30 dim char LSN(134),LSR(134),SSN(39),SSR(39)
40 dim char NUM(149),RDAT(1),WDAT(4),CFONT(2303)
50 dim int FONT(14),CLDAT(1)
60 dim int NPAT(55)={
70     24,21,27,21,24,27,11,17,31,37,
80     11,17,24,31,37,11,14,17,31,34,
90     37,11,14,17,22,31,34,37,11,14,
100    17,22,26,31,34,37,11,13,15,17,
110    24,31,33,35,37,11,13,15,17,22,
120    26,31,33,35,37,1318
130 }
140 J=0
150 for I=0 to 54:J=J+NPAT(I):next
160 if J<>NPAT(55) then {
170   print"配列 NPAT の内容に誤りがあるようです":end
180 }
190 screen 1,1,1,1
200 w=fopen("DATA_CL.DAT""c"):r=fopen("FONT.dat","r")
210 w_PR=fopen("CARD.FNC","c")
220 fread(CFONT,554,r):fwrite(CFONT,554,w_PR)
230 for I=0 to 553:CFONT(I)=0:next
240 for I=0 to 59:fwrite(CFONT,2304,w_PR):next
250 /*
260 fill(0,0,56,30,1):fill(16,0,40,30,5)
270 for ST=0 to 3
280   fread(FONT,15,r)
290   for I=0 to 14
300     line(ST*16,I,ST*16+8,I,15,FONT(I))
310     line(ST*16,30-I,ST*16+8,30-I,15,FONT(I))
320   next
330 next
340 fill(0,40,31,56,1):fill(8,40,21,56,5)
350 for ST=0 to 1
360   fread(FONT,8,r)
370   for I=0 to 7
380     line(ST*16,I+40,ST*16+15,I+40,15,FONT(I))
390     line(ST*16,56-I,ST*16+15,56-I,15,FONT(I))
400   next
410 next
420 w=fopen("DATA_ST.DAT","c")
430 for X=0 to 3
440   get(X*16, 0,X*16+8,14,LSN):fwrite(LSN,135,w)
450   get(X*16,16,X*16+8,30,LSN):fwrite(LSN,135,w)
460   get(X*8 ,40,X*8+4 ,47,LSN):fwrite(LSN, 40,w)
470   get(X*8 ,49,X*8+4 ,56,LSN):fwrite(LSN, 40,w)
480 next
490 fclose(w)
500 wipe():fill(0,0,112,79,1):fill(0,20,112,59,5)
510 for I=0 to 6
520   fread(FONT,10,r)
530   for J=0 to 9
540     line(I*16       ,J   ,I*16+15,J   ,15,FONT(J))
550     line(96-I*16+15,19-J,96-I*16,19-J,15,FONT(J))
560     line(I*16       ,J+20,I*16+15,J+20,15,FONT(J))
570     line(96-I*16+15,39-J,96-I*16,39-J,15,FONT(J))
580     line(I*16       ,J+40,I*16+15,J+40,15,FONT(J))
590     line(96-I*16+15,59-J,96-I*16,59-J,15,FONT(J))
600     line(I*16       ,J+60,I*16+15,J+60,15,FONT(J))
610     line(96-I*16+15,79-J,96-I*16,79-J,15,FONT(J))
620   next
630 next
640 w=fopen("DATA_NM.DAT","c")
650 for J=0 to 3
660   for I=0 to 12
670     get(I*8    ,J*20   ,I*8+4    ,J*20+9 ,NUM)
680     fwrite(NUM,150,w)
690     get(107-I*8,J*20+10,107-I*8+4,J*20+19,NUM)
700     fwrite(NUM,150,w)
710   next
720 next
730 fclose(w)
740 wipe():cls:print"ただ今データの展開を行っています"
750 locate 0,1:print"    /5254"
760 w=fopen("DATA_EC.DAT","c")
770 for CT=0 to 1599
780   fread(RDAT,2,r):WDAT(0)=RDAT(1)
790   for I=0 to RDAT(0):fwrite(WDAT,1,w):next
800   locate 0,1:print right$("    "+str$(CT),4)
810 next
820 fclose(w)
830 w=fopen("DATA_HF.DAT","c")
840 for CT=1600 to 5254
850   locate 0,1:print right$("    "+str$(CT),4)
860   fread(RDAT,1,r):EX(RDAT(0)):fwrite(WDAT,2,w)
870 next
880 fread(CFONT,44,r):fwrite(CFONT,44,w_PR)
890 fcloseall()
900 cls
910 r=fopen("DATA_HF.DAT","r"):w=fopen("DATA_AN.DAT","c")
920 w=fopen("DATA_AN.DAT","c")
930 for I=0 to 11
940   fill(0,0,30,84,1):fread(CFONT,609,r)
950   put(1,1,29,42,CFONT):REV()
960   get(0,0,30,84,CFONT):fwrite(CFONT,1318,w)
970 next
980 fcloseall()
990 w=fopen("DATA_CL.DAT","c")
1000 NP=NPAT(0):NPAT(0)=0:fwrite(NPAT,55,w):NPAT(0)=NP
1010 for I=0 to 2:fwrite(NPAT,55,w):next
1020 fclose(w)
1030 w_PR=fopen("CARD    .FNC","w"):fseek(w_PR,554,0)
1040 r_ST=fopen("DATA_ST.DAT","r")
1050 r_NM=fopen("DATA_NM.DAT","r")
1060 r_AN=fopen("DATA_AN.DAT","r")
1070 r_CL=fopen("DATA_CL.DAT","r")
1080 ETC(1):CD_WRITE()
1090 for CT=0 to 3
1100   SUT_READ()
1110   for I=1 to 10
1120     PAPER()
1130     for J=1 to I
1140       fread(CLDAT,1,r_CL):PUTST(CLDAT(0),I)
1150     next
1160     CD_WRITE()
1170   next
1180   for I=0 to 2
1190     PAPER():AN_READ():CD_WRITE()
1200   next
1210 next
1220 ETC(2):CD_WRITE()
1230 fcloseall()
1240 fdelete("DATA_ST.DAT"):fdelete("DATA_NM.DAT")
1250 fdelete("DATA_EC.DAT"):fdelete("DATA_HF.DAT")
1260 fdelete("DATA_AN.DAT"):fdelete("DATA_CL.DAT")
1270 width 96
1280 end
1290 /*
1300 func CD_WRITE()
1310 dim char A(2303)
1320 get(0,0,47,95,A):fwrite(A,2304,w_PR)
1330 endfunc
1340 /*
1350 func SUT_READ()
1360 fread(LSR,135,r_ST):fread(LSN,135,r_ST)
1370 fread(SSN, 40,r_ST):fread(SSR, 40,r_ST)
1380 endfunc
1390 /*
1400 func PAPER()
1410 dim char N(149)
1420 fill(0,0,46,94,15)
1430 pset(0,0,0):pset(46,0,0):pset(0,94,0):pset(46,94,0)
1440 put(2,14,6,21,SSN):put(40,73,44,80,SSR)
1450 fread(N,150,r_NM) :put( 2, 2, 6,11,N)
1460 fread(N,150,r_NM) :put(40,83,44,92,N)
1470 endfunc
1480 /*
1490 func PUTST(A,B)
1500 int X,Y
1510 if A>10 then {
1520   X=A/10*9+1:Y=(A mod 10)*10
1530   if Y/10=2 and B<>10 then Y=Y+5
1540   if Y/10=6 and B<>10 then Y=Y-5
1550   if Y>41 then {
1560     put(X,Y,X+8,Y+14,LSN)
1570   } else {
1580     put(X,Y,X+8,Y+14,LSR)
1590   }
1600 } else ETC(0)
1610 endfunc
1620 /*
1630 func AN_READ()
1640 dim char A(1317)
1650 fread(A,1318,r_AN):put(8,5,38,89,A)
1660 endfunc
1670 /*
1680 func ETC(N)
1690 int r,I
1700 dim char C(2303)
1710 r=fopen("DATA_EC.DAT","r"):fread(C,729,r)
1720 if N=0 then put(10,22,36,75,C):fclose(r):return()
1730 for I=1 to N:fread(C,2233,r):next
1740 put(0,0,46,94,C)
1750 fclose(r)
1760 endfunc
1770 func REV()
1780 int X,Y
1790 for Y=0 to 41
1800   for X=0 to 30
1810     pset(30-X,84-Y,point(X,Y))
1820   next
1830 next
1840 endfunc
1850 func EX(DAT)
1860 int D1,D2,D3,D4
1870 D1=DAT/64:DAT=DAT mod 64:D2=DAT/16:DAT=DAT mod 16
1880 D3=DAT/4:D4=DAT mod 4
1890 WDAT(0)=CV(D1)*16+CV(D2):WDAT(1)=CV(D3)*16+CV(D4)
1900 endfunc
1910 func CV(DAT)
1920 switch DAT
1930   case 0:return(1):break
1940   case 1:return(5):break
1950   case 2:return(13):break
1960   case 3:return(15):break
1970 endswitch
1980 endfunc
```


▶留年および浪人がすでに決まってしまった皆さん，おめでとうございます。他人からなにかいわれたら，開き直って「うらやましいだろう」と言ってやりましょう。
森川　昭夫 (21) X68000, X1C　東京都

●特集　BASICプログラミング

BASIC PROGRAMMING

パーソナルコンピュータの未来。高速処理，誰にでも触れるわかりやすいユーザーインタフェイス，ソフトウェアはもっともっと進化していかなければならない。しかし，そういったコンピュータの前に座る人間はどの程度進化していくのだろうか。
市販ソフトが多様化したことにより，プログラムを組む人は少なくなったといわれる。パソコンの用途が多様化しているのだ。これは歓迎すべきことでもあろう。
半面，その結果として，いまや多くのパソコンで「コンピュータと対話すること」自体が難しくなっている。かといって，プログラミングを必要とし，かつそれを目的にする人たちがいなくなることはありえない。いまもBASICはそういった人とパーソナルコンピュータを結ぶ糸口なのだ。

CONTENTS

BASIC

入門者のための

X-BASICの心得

Nakano Shuichi
中野 修一

BASICとC言語の融合体ともいえるX-BASIC。より高級言語的な姿に，プログラミング入門者ならずともわかりにくいところがあるのではないでしょうか。ここではX-BASICを使ったプログラミングの考え方をみてみましょう。

BASICが難しい?

DIME誌によるとBASICをマスターできるのは入門者の1000人にひとりなのだそうだ。ふーむ。経験からいってアセンブラが使える人はせいぜいBASICユーザーの数パーセントだろうなぁ。すると，22万台出荷のX1シリーズでアセンブラを使える人は片手の指くらい，ようやく10万台のX68000にいたっては妖怪人間の片手でまにあってしまう……。

確かに，とある日本の普通のパソコンではディスクBASICはたいてい封も開けられず眠りにつくことになっている（賢い人はプロテクトシールをはずしてデータディスクにする)。

コンピュータのアイデンティティが「プログラムできる機械」であった時代は忘れられようとしているのだろうか。

Beginners' BASIC

ありがちなパターン。X68000を買ったらバリバリプログラムするぞ！　と志に燃えていても，いざ買ってみたら，ついついゲームをしてしまう。アンケートはがきを見ていてもよく目につく。

確かに実際，X68000ユーザーはよくゲームをする。若い衆だけでなく，一見どこにでもいそうなビジネスマンが平気でグラディウスを1周したりする。X68000は触っているだけでも楽しい。といってもゲームばかりしているわけじゃない。ユーザーの6人にひとりは重さ5kgのCコンパイラを買っている。……こんな人は，すでに世間ではマニアと呼ばれX68000ユーザーの中核をなしている人たちでもある。

いまはゲームに興じている人も，もっといろいろなことがしたいと思っているはずだ。X68000はゲーム以外にも，いろいろ“遊び方”のあるパソコンだ。君のパソコンはプログラムを待っているぞ。

それとは別に，初めて買ったパソコンがX68000で……という場合，BASICはもちろん，ディスクのフォーマットだって教えられなきゃわかるはずがない。それでも，グラフィックツールで作ったデータをもっと役立てたいとか，自分でもゲームを作ってみたい，と思っている人は多い。

X68000のシステムディスクにはDOS，スクリーンエディタほか各種コマンド群，BASIC と必要なものはほとんど揃えられている。実はやろうと思えばできる環境がちゃんとあるわけだ。でも，とっかかり口がわからない。そんなとき頼れるのはやはりBASICしかない。

X-BASIC は単体ではさほど強力ではない。MZ-2500のBASIC-M25やPC-88VAのほうが強力なBASICという感じがする。が，柔よく剛を制す。外部関数で強化されたりコンパイラとセットになるとX-BASICも途端に強力なシステムとして生まれ変わる。実に拡張性に富んだ言語だ。

実際にBASICとコンパイラで作られたゲームを見ても，ハードウェアの表現力を生かすだけの根気さえあればそこそこのものは作れるということがわかる。

問題はいかにBASICを使いこなすかだ。文法を理解すること以前にしなければならないこともある。パソコンの感覚に慣れてしまうことだ。昔は，私も“A=A+1”は不可解だったし，GOTOがまさか“go to～”の意だとは思わなかった。ましてリターンキーの存在理由など思いもつかなかった。これからの特集を読むうえで最低限マニュアルの冒頭くらいは軽く読んでおいてほしい。

プログラムの考え方

コンピュータというのは，人間がやりたいと思ったことをなんでもやってくれるわけではない。人間が指示したことをそのとおりにやるだけだ。つまり，コンピュータになにかさせようとすると，その過程で必要なことを前もってすべて指示しておかなければならないことになる。これが，すなわちプログラムだ。

どんなコンピュータも最終的にはマシン語で動作する。でもいちいち全部をマシン語で組むのは面倒なので，いろいろなところで使う決まりきった処理はあらかじめ用意しておいて，必要なときに呼び出して使う。それでもわかりにくいので，人間にわかりやすい記述で書いても，ちゃんと必要な処理を行ってくれるものが出てきた。これがプログラム言語だ。

BASICは自分の中にたくさんのマシン語処理ルーチンを抱え込んでいる。BASICプログラミングとは，人間がそれらをどういったとき，どういうふうに呼び出すかを指示することにほかならない。

まず，X-BASIC では関数というものを理解してないとマニュアルに書いてある命令の書式さえわからない。

関数というのは，

$y=f(x)$

だ。f( )という箱にxを投げ込むとyが出てくるという機構。これは数学の時間に習ったとおりだ。ひとつのxに対して必ずひとつのyが決まればよい。別に規則性などはなくてもいい。この場合はf( )が関数，xが引数，yが戻り値と呼ばれる。マニュアルに書いてある命令はほとんどが関数だ。

引数と戻り値には「型」がある。型にはchar(小さな整数)，int(大きな整数)，float(実数)，str(文字列)があって，要するにBASICに用意された処理ルーチンに都合のよい呼び出し方で呼び出してやらなきゃいけない。変数や定数はふつう黙っていればintとみなされる。

数学の関数は数値を違う数値に変えるだけだが，コンピュータ上で「関数」といった場合は数値を返す以外にいろいろと「副作用」を起こすことが多い。つまり，文字の色を変えたり，FM音源を鳴らしたり，円を描いたりといったことだ。

つまり，関数を使うというのはこの副作

▶「アルガーナ」をやってみたのですが，主役キャラが女ってのはやりにくいものですね。女になったつもりにはなれなかったので，おかまになったつもりでプレイしました。
伊藤　雅彦（20）X1turboZ II　東京都

用を目的に使うことが多い。甚だしい場合は副作用だけで戻り値がない場合がある。円を描くcircleなどがそうだ。これは関数の定義とは相容れないのだが呼び出し方は関数と同じなので関数と呼ばれる。

たとえばpaint( )。私たちはペイントルーチンのアルゴリズムを知らなくても画面を思いどおりに塗ることができる。使い方(引数，戻り値，副作用)さえわかっていれば関数の中身は知らなくてもいい。

もし，これからやりたい処理にピッタリの関数がたくさん揃っていれば，プログラムは非常に簡単かつ，すっきりできるだろう。でもちょうどいいのがみつからない。なら自分で作ってしまえ！　というのがユーザー定義関数だ。関数の中身はBASICで書ける。そして書いてしまえば，組み込みの関数とまったく同じように扱うことができるのだ。

X-BASIC では，関数を作り，組み合わせていくことがプログラミングの基本スタイルになる。どんなものを与えたら，どんなものが返ってきて，どんな副作用が起こってほしいのかを考えながらプログラムを組むことになるだろう。

「構造化」というもの

プログラムに必要な処理は大別すると，処理の流れが単に進むだけの「順次処理」，枝別れする「条件分岐」，そして「繰り返し」の3種類になる，とする。これを明確に意識するとプログラム全体の構造がわかりやすくなる。構造化プログラミングとはプログラムの無駄な流れを省いて，アルゴリズムの読みやすいプログラムを書こうということだ。

昔のBASICやFORTRAN ではプログラム記述の際の自由度が低く，プログラムを作っているとどうしてもゴチャゴチャしたものになりやすいのだ。いまのBASICにはプログラムの動作にはあまり関係ないが，人間が読むときにわかりやすいようにするための命令，構造化命令が加わっている。

無条件ジャンプはやめよう。

サブプログラムでは「入り口」をひとつ作って,「出口」をひとつ作る。途中で必要な処理が完了しても，ちゃんと出口までいって処理を抜けるようにする。サブプログラムに限らず「入り口ひとつに出口ひとつ」が原則だ。

構造化と似たような言葉に「モジュール化」という考え方がある。汎用的に使える部品を組み合わせて，効率のよいプログラミングを行おうというわけだ。

X-BASICでも関数単位にsave@ しておくことである程度のモジュール化をすることは可能だ。あまり使わない関数の仕様などはたいてい忘れてしまうので，頭書きに注釈で処理内容，パラメータ，副作用，要求されるグローバル変数などをまとめておくといいだろう。

が，よほど大掛かりなプログラムを作るときでもないとモジュール化まで考える人はいないだろう。どのあたりでモジュールにまとめるかを考えるだけでも結構面倒なものなのだ。

特に初心者のうちは，変にラクしようと思わずに，同じルーチンを作り直してみるほうが収穫は大きいと思う。

知恵袋

最後にBASICでの小技，基本技をざっと紹介しておく。X-BASIC 入門者は参考にしてほしい。

＊　＊　＊

X-BASIC 入門者が最初に行うべきことはBASIC起動時の状態を設定することだ。chdirなどの基本的なコマンドに省略形がないときは，

key 11, “chdir@@”

のようにファンクションキーを設定しておくプログラムを作って起動時に実行させるようにする。

ファンクションキーも一例だが，BASICはプログラミングのための「環境」を備えた言語だ。エディットのために便利な機能を備えている。

まず，コントロールコードというものを覚えよう。ctrl-Eやctrl-Z は知っているといないとでは大きな違いだ。

さらにマニュアルには書いていない機能も知ってないと損だ。たとえば，

l. .

はどのような動作をするのか？　2つ目のピリオドにはBASICがプログラム解釈中の行番号が入る。エラーで実行が止まったときは，実行の停止した行番号，エディット時には最後に入力した行番号になる。たとえば，1050行でエラーが出たときに，

l. . －950

とすると950行からエラー発生行のリストが見れる（めったにしないな)。

＊　＊　＊

エラーで実行が止まっても，プログラムの先頭で宣言した変数は参照できる（プログラムの変更をしないかぎり)。

また，ブレイクキーで実行を止め，ある変数を適当に書き換えてcontで継続することも可能。

＊　＊　＊

文字列のサーチは非常に多用されるコマンドだ。

search “文字列”

のようにすればプログラム中から文字列を含んだ行を探せるが，打ち込むときは，

se.” 文字列

で十分だ。X1ほどでないにしても省略形は許されており，文字列の終わりのダブルクォーテーションも省略できる（プログラム中では間違いの元にもなる)。

＊　＊　＊

リストの表示で一時停止させるときなどのctrl＋Sにも慣れただろうか？　長いリストで間違って違うキーを押したときなどに不快な目に会いたくないときはシフト＋ブレイクキーを押すようにする。ちょんと押すのではなく，ぐっと押さえつけるのがコツ。シフト＋ブレイクはキーバッファをクリアしてくれるらしい。

＊　＊　＊

文字列関係の関数はひとつの式で一度にいくつも書かないほうがいい。

＊　＊　＊

チャイルドプロセスも強力。これに慣れるとBASICからエディタを起動したり，実行時間がかかるものはrun する代わりにコンパイルして実行するということが当たり前のようになる。メモリさえ十分に増設すればBASICからGCCを BASICコンパイラ代わりに起動してもびくともしない。昔はX1の320KバイトEMMが88,000円だったのに，いまなら同じ値段で4MバイトのRAMボードが買える。時代は変わった。

＊　＊　＊

行番号のないファイルはload@＋ファイル名＋行番号で行番号位置から重ねてロードされる。最終行以外では十分注意するように。

これを使うと過去に作ったプログラムの使い回しが楽にできる。しかし，使い回される頻度の高いスプライト定義やFM音源の音色定義のように，行番号を取ると配列の数字がむき出しになる場合には「行番号のあるファイルです」と怒られることになる。これを避けるため，各行の最初の数字にわざわざ“＋”などの符号をつけたりするのも生活の知恵だ。DEFSPTOOL.BASはすぐに書き直そう。

＊　＊　＊

ではよいプログラミングを。

▶1月号の「オペレーティングスタイルの研究」を読んで以来，VS.X に再び戻りつつあります。VS.X の画面レイアウトの美しさは，ほかに類を見ないと思います。
山本　誠司 (24) X68000ACE　東京都

[特集]
BASICプログラミング

潜入! バグ対策24時間, X-BASICはいま……!

プログラムを組んでいると必ず出会うのがバグ。また，Oh!Xの掲載プログラムを打ち込んでも一度ですんなり動いたという人は少ないはずです。初心者にありがちなミスをチェックする方法をざっと紹介してみましょう。

Komura Satoshi
古村　聡

X68000の場合，ゲームやワープロをやっていて「違うっ！　断じて違うっ！（古いなー)」とか「いいなー，こういうの作ってみたいなー」と思う人が多いのか，わりと誰しも「プログラム作ってみたいなー」と思っているみたいなんですね。で，自分でプログラムを作るために，さしあたってはBASICに手を染めて（でも，X-BASICというやつはコンパイルもできたりとあなどれないBASICだったりする）みるわけなんですが……。

X-BASICを起動するにはコマンドモードでbasicと打つか，VSモードでbasic2，basic.xとダブルクリックすればいいんだな，ふんふん。プログラムを1行打ったらリターンキーを押すと。よしそれじゃ，マニュアルのサンプルをひとつ打ってみるか。それともOh!Xに毎月連載の「ショートプロぱーてぃ」に載っているプログラムとかを打ち込んでみようかな。カタカタと。よし，runを打つと……。がーん，エラーだ。

まぁ，初心者でなくたって，たいていの人は1パツで動いちゃうことなんてほとんどないはずですし，さらに打ち込みでなく自作のときなんかはもう出る出る，エラーが。ま，打っているのが人間なんだから(まして自作の場合は創作活動という人間にしかできない尊い作業をやっているわけだ）多少の間違いはとーぜん，したがってエラーだってとーぜん，なんですよね。ああ，それなのにそれなのに，なぜか「やーめた」といってここ（あるいはプログラム何本目かのここ）で投げてしまう人がわりと多いみたいなんですね。

うーむ。X-BASICってちょっと変わったBASICだから，X-BASICが初めてのBASICという人だと相談しようにもX68000を持ってる人にしか相談もできない，なにを作ろうにもBASICのエラーの対処の方法がわからない。いったんバグったらほとんどパニックになる。それはなんとなくわかる。でもでも，いいプログラムに1本でも多くなってほしいと願う私としてはどうにか立ち直ってほしいなー，などと思うわけです。

で，今回はそんなしつこいバグ対策として緊急企画「潜入！　バグ対策24時間，X-BASICはいま……！」をお送りしたいと思います（ちょっとオーバーかな……)。さぁ，ご期待。

作れば当然バグは出る

まず，初心者が陥りやすいパターンとして，

　　バグ取りがバグ

というのがあります（ちなみにこれ，ミイラ取りがミイラのパロディね。ああ，ギャグの解説はむなすい……)。要するに,バグを取っているつもりで実は正しいプログラムだったところにさらにバグを加えてしまうということなわけです。なーんで，こんなことになってしまうかというと，

1）　これどうしたらいいのー!?

デバッグの定石がわかってないってやつですが，これは当然でしょうねー。プログラムの組み方，デバッグの定石が初めからわかってる人だったら初心者じゃないってばって。でも大丈夫。この続きをずーっと読んできゃあなたもベテランプログラマ（……になれるといいね，と逃げる私)。

2）　あーあ，打ち込むの疲れたー

コーディングしたプログラムをいっぺんに全部入力してしまった，というパターン。なにせキーボードを叩くというのは疲れますから，打ち間違い，タイプ抜けがゴロゴロ出てきます。というかいくらプログラムが組めるようになった人でもそこそこプログラムが大きくなるとタイプミスは結構するもんです。が，やっぱり初心者の人のほうがエラーの数も多いうえに1つひとつのバグもなかなか取れなくてそのうち投げちゃうわけです。プログラムは少しずつ正確に打ちましょう。

3）　え，いっぺんに打っちゃった

さっきのと似てますが，こんどはディスクをケチっちゃったってやつ。これは自作のときにありがちですが，バグ取りがバグになって，そのバグがバグを生み，さらにそのバグがバグを……と，「おーい！　なにやってるんだぁ!?」級の事態になることもたまにありますからねー。

そういう事態にならないようにまずプログラムをセーブしておくのです。で，プログラムをデバッグして，あ，違ったかなと思ったらさっきセーブしたプログラムをロードしてしまうのです。これだとわりといきあたりばったりにデバッグできますからねー(初心者にこんなこと教えてどーすんじゃ)。ちなみにちょっと大きなプログラムを開発するときなどになるとフロッピー中にそのプログラムがいっぱいなんていうのはまだかわいいほうで，似たようなプログラムが入ったフロッピーが何枚も何枚もあるなんていうのはよくあることです。フロッピーはまとめ買いして湯水のように使いましょう。

まずは一般的な注意

で，今度はエラー対策の定石のほうにいきます。X-BASICもBASICですからほかのBASICでも通用する定石というのがあります。そういう手は，実は，一般的な言語ならCとかPASCALとかでもそこそこ通用するので結構お徳用なんですよ。さー，どなたもどなたもお立ち会いっ。

1）　げげっ，文法エラーが出ちゃった！

ごく普通の打ち間違いっていうか，いわれたとおりに打ち足すやつ，早い話がほかのBASICでいうSyntaxErrorってやつがあります。この文法間違いはX-BASICのほうもだいたい心得たもんで，「，がありません」とか「whileに対応するendwhileがありません」とか「文が実行できません」とかいってきます。

```
文が実行できません……10行
10 id g=1 then print 20
     ^
```

その行を表示して，そのうえだいたいの場所まで表示してくれます。うーむ，賢い。ただ，残念なことにこのエラー位置表示もほとんどの場合間違っている箇所を指してくれません（しかたのないことなのはわかるんだけど)。BASICは命令を順番に解釈していき，つじつまがあわなくなったらエラーを出します。“＾”マークは「つじつまがあわなくなった場所」に出るので，その原因は直前あたりにあることが多いんですね。

さっきの場合も間違っているのは“＾”の指しているスペースではなくてdです。それでもこの手のエラーはそのあたりの数字をチェックするか，エラーメッセージどおりにとか，endwhileとかを付け加えたり書き換えたりするだけですからねー。楽勝，楽勝（でも，セーブはちゃんとしてくださいね)。

2） なにこれ！ ちゃんと動かない！

おいおいこの行のどこがエラーなのー(表示された行は間違えてない)，とか，

引数が規定の範囲を越えています

○番目の引数が無効です

なんていう変数の内容がおかしいという場合，ふつうのBASICだとillegal function callになるやつらがいます。

こいつらの場合，せっかくBASICがエラーメッセージを出してくれたのにぜーんぜんあてにならなかったりするので，事態はゲリラ戦になってしまったりと結構ややこしかったりします。

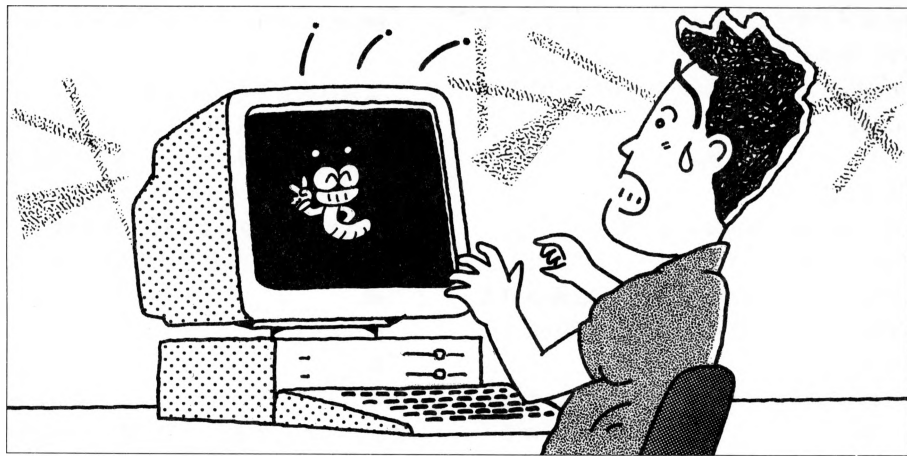
たとえば質問電話でこんなのがあったそうです。

「○○行でエラーが出るんですが，いくら見直してもその行は間違ってません。その行を取るとエラーは出なくなったんですが，やっぱり動作がおかしいんです。このプログラムにバグはないでしょうか？」

「……」

たとえば整数の範囲だけどパレットコードとしては大きすぎるなんて場合。リスト1なんかを実行してみるとこれもBASICは40行のエラーとして出してくれますが実はこいつがヌレギヌで，実は虫が潜んでいるのはお隣の35行（この悪党ぉっ！）なんです。

え，なんで，まずいかって？ パレットコードに使えるのは65535まででしょ。でもINTは2147483647まで許してしまうんで，大は小を兼ねるのに逆は必ずしも真ならずということです(どういう日本語だ……)。

さて，このようなゲリラ戦を展開するこの手の虫の居所に当たりをつける方法をつらつらっと書いてみます。

1） 1ステップ実行

2） 変数をprintしてみる

3） ダミーの変数を作る。

こんなところでしょう，代表格は。

まず最初のは，プログラムの間違えてそうなあたりの行番号を取り除いたり，マルチステートメント（:）でつながっている部分を1個1個切り離して実行して様子を見てみるというやつです。プログラムを組むこと自体が1つひとつの動作をあわせていったものなのでそれをバラして見ていけば自分の思い違いを見つけるにはいちばん！というやり方です。

2番目は1番目をちょっと手早くやる方法。1ステップずつやるのは確実に見つけたいときにはいいのですがこれをやると時間がかかってしょうがないんですね。で，とりあえずある変数の変わり方だけ，ぱーっと見ていこうってのがこれです。

具体的にはプログラム中の適当なところにprint○×というのをつけてみるということになります。これだと実際にプログラムを動かすのとそんなに変わらなく実行できて時間の節約になりますが，プログラムの動きの一部しかわからないので一度ではどこが悪いのかわからないことも多々あります。たとえるなら2)は体温計，1)は実際にメスで細かく切ってみるという感じ（グ，グロテスクな……）ですかねー。

最後のは，そうですねー，バリウム飲んで胃カメラで撮るみたいな感じかなー。要するにプログラムがどのへんの行を動いているのか見る方法で，ダミー変数を作ってやって，プログラム中のいろんなところでダミー変数の内容を変えてみるのです。

たとえば（リスト2）のようにしてやれば何行めまで進んだのかが一目瞭然でしょ？ま，これはちょっとオーバーですけど，怪しそうなところでいろいろダミー変数を変えてやるってのはメンドクサイけどなかなか強力なワザなんです。

もちろんこいつらを単独でなくても組み合わせるのってーのもありで，変数表示であちゃ！ と叫び，ダミーをつけてはムムムとうなり，最後に1ステップずつサクサク刻んでしまうなんてーのは，プログラムする姿勢としてはとってもよいのじゃないかなーなんて，私は思います。

Oh! X-BASIC……

ところで，X-BASICをほかのBASICと比べてみると「関数が作れる」「ローカル変数がある」などの特徴があります。ま，これはこれでなかなか便利な機能なんですけど，なにせがいままでのBASICにはそういうものがなかっただけにこれがらみのエラーはいままでのBASICを知ってた人にとっても，とーっても（しゃれじゃないよ）

リスト1

```
10 screen 1,3,1,1
20 x1=0:y1=0
30 x2=1024:y2=1024
35 c=65536
40 line(x1,y1,x2,y2,c)
```

リスト2

```
 5 int x1,y1,x2,y2,tron
10 screen 1,3,1,1:tron=10:print tron
20 x1=0:y1=0:tron=20:print tron
30 x2=1024:y2=1024:tron=30:print tron
40 c=65536:tron=40:print tron
50 line(x1,y1,x2,y2,c):tron=50:print tron
```

▶TETSUってそんなにまずいですか？ 私はそーでもないと思います。
平野 直樹 (19) X68000ACE-HD, FM-77AV, MSX2 神奈川県

メンドクサイことになってしまったりします。そりゃそうですよね。いままでなかったんだもん，マイクロソフト系BASIC(注1)には。

で，いままで遭遇したことのないいちばん大きなヤツに，

1）ロ，ローカル変数はどこ!?

というのがあります。ローカル変数というのはfunc～endfuncのなかで定義された変数でそのなかでしか使えない変数のことなんですが……。

ほほう，ローカル変数か。じゃ，ちょっと使ってみよう。ピッ！ おっと，エラーか。じゃ，さっそく体温計の方法ってことでさっき宣言したローカル変数を見てみるか。print lx。え，なにーっ！

変数は宣言されていません

だとーっ！ 馬鹿な，このプログラムは死んでいるのかーっ！ ……違いますって。実にめんどくさいことに，ローカル変数っていうのはブレイクキーを押したりエラーで止まったりすると変数が消えてしまうんですよねー。関数に数値や文字列を渡すときに使う引数もそうで，たとえば（リスト3)のようなリストを打ったとします。すると65行で文法エラーで止まるわけですけど，例によってlxを見てみようとして，

リスト3

```
10 int a,c,x
20 c=5:x=10
30 a=b(x)+c
40 print a
50 end
60 func int b(lx)
61 lx=lx+1
65 ここがエラーだ
70 return(lx*10)
80 endfunc
```

リスト4

```
10 int a,c,x
20 c=5:x=10
30 a=b(x)+c
40 print a
50 end
60 func int b(x)
61 x=x+1
65 ここがエラーだ
70 return(x*10)
80 endfunc
```

リスト5

```
10 int a,c
15 dim int b(30,30)
20 c=10
30 a=b(1,2)+c
40 print a
45 end
50 func int b(e,f)
60 return(10)
70 endfunc
```

print lx

としても，

変数は宣言されていません

などとここでも血も涙もないエラーメッセージが出てしまうわけです。さらに混乱するパターンに（リスト4）のようなものがあります。これもさっきと同じく65行で止まるんですが，xの内容を見てみるとあーら不思議，

print x

10

と表示されてしまうんですよね。そう，65行で止まったにもかかわらずその直前の60，61行で使われたローカル変数のほうのxじゃなくてもっと前の10，20行のグローバル変数のxが表示されてしまっているんです。

このくらい短いプログラムならどうってことないですけどこれが1000行とか2000行とかの長いプログラムだともう，パニックに陥ること，間違いなし！ この手のミスはなかなか見つかってくれませんからねー。なぐってやろうかと思ってしまいます。

でもですねー，はっきりいって，ローカル変数が消えちゃうっていうのはどうしようもない半分宿命みたいなもんなんです，はっきりいって。ローカル変数が使えるだけでももうけものって話もあるし。

じゃぁ，こいつをどうするか。自作の場合だったら，とりあえず変数さえ見えればなんとかバグ取りできそうって状況なら，思いきってぜーんぶグローバル変数にしちゃうのも手です。プログラムが汚くなっちゃうかもしれないし，リカーシブコールもできなくなっちゃうけど動けばいいやっていう場合にはベストです。コンパイルするときにローカル変数の大きさを気にしなくてすむので（コンパイルはローカル変数の大きさが32Kバイト以上になるプログラムはできないんですよね。めったにそんなに大きくならないけど……）一石二鳥な方法です。

でも，どうしてもモジュール化してあとでもう1回このルーチンを使いたいとか，再帰呼び出しのやり方を憶えたいんだい！っていうこともたまにはあるんじゃないかと思います。こんなときはとにかく関数をできる限り小さくして1個1個動かしてみてバグがないことを確かめる！ 小さいプログラムならデバッグも楽ですから，バグの芽は小さいうちにってなもんです。それとさっきの胃カメラの方法で「怪しそうなあたりで変数や行番号がわかるようにしておく（TRONがあれば……)」というのも結構有効です。そして，まずいなっと思ったらブレイク！ これが切り札です。え，それでも取れなかったらどうするかって……うーん，寝てしまえっ！（じょーだんですってば）

クサいバグはモトから断たなきゃダメ！

さて，それからX-BASICでは関数がらみのエラーも強敵です。

「関数は定義されていません」なんていうのが代表的ですが，エラーメッセージが出ない配列関係のバグもしつこいです。たとえば（リスト5）みたいなプログラムがあります。この場合は答えはいくつになるでしょう？ 20!? おおはずれーっ！ 10になるんです。30行で関数を呼び出すつもりが間違えて配列を呼び出しちゃった。というバグなんですが，はっきりいってこれはX-BASICが悪いっ！ だっていきなり配列と関数の呼び出し方が同じなんですよ。ひどいと思いません？

まぁ，いまさらそんなこといってもしょうがないんでとりあえず対策なんですが，こういう場合は変数名や関数名を変えてみるっていうのはどうでしょ？ 私なんかはこういうエラーがないように配列を使うときは名前の後半にaryとつけるようにしてます。たとえば答えの入る配列だったらansary（もちろんanswer's arrayの略）とかね。変数や関数の名前に意味を持たせるっていうのはわりとどの言語でも使える鉄則ともいえるんで，やってみてそんはないんじゃないかと思います。ほかの人にも読みやすくなるし。

そうそう，名前が長くなるとこっちがスペルミスしそうだなーなんて思ってるそこのあなた，そんなこたぁない。人間なんだかんだで憶えてるもんですよ。いやいやそれでもと思うんなら，さらに「名前は絶対6文字になるようにする」とか自分で決めとけば，もう完璧です。ついでに私のやる手口をちらっと教えちゃうと一度プログラムリストをプリントアウトして見てみるというのをよくやります。これはいいですよぉ。whileに対応するendwhileとかifのあとの｛ ｝とかの対応を赤鉛筆で矢印で「これとこれ」って一発でわかるようにするんです。

紙に書くっていうのは特にマシン語でのプログラミングに威力を発揮するみたいで，スタックにひとつ積んだら＋をひとつ，ひとつ減らして－1なんてやってスタックの対応なんかを書いておくと効果はテキメンです。もっとも，この方法を使うにはプリ

▶MZ-700でスペハリができるのであれば，X1turboでアフターバーナーができるかもしれない。X1turboに不可能はない!? 塩瀬 多聞（17）X1turboZ II 神奈川県

ンタを買ってこなきゃなりませんけど……。プログラムっていうのも結構お金のかかるもんなんですよね。

LIVE派のBASIC

そうそう，最近はLIVE打ち込み派の方が増えているみたいですね。で，もう当然音楽演奏関係のエラーメッセージとのおつきあいが多いはずのこの方々の中には気づいてる人も多いと思いますが，「m_trkで出てくるエラーメッセージはあてにならない」ことが多いんです。music関係の命令というのは，

　　文字列＝“音楽関係の命令～”
　　m_trk(1,文字列)
　　m_play( )

というふうに三段逆スライド方式（ああっ，また誰も憶えてないギャグをっ！）になっているのが普通です。が，まずいことに，たとえば（リスト6）みたいに文字列に入れるものを間違え（つまりまずいのは25行なわけだ）てもエラーはm_trkのある30行で出たりするんですよね。で，さらにまずいことにミュージックプログラムでは文字列が配列になっていたり，m_trkをforループでくり返し実行してしまっていたりすることもよくあったりします。

ということで，音楽関係のバグ取りの必勝法は，音楽関係の命令を使ってエラーが出たら，それを設定している部分を探す，これでいくしかないことになります。m_trkでなけりゃその前の文字列の定義を確実に探っていく，もうどんどん遡っていくしかありません。BASICでのバグ取りの知識をフルに生かしてfor文のループ回数やらなんやらを見て逆探知。根気と要領です。

……教訓。音楽しかやらなくてもBASICと体力作りはひととおりやっておきましょう。

バグ取り最終奥義っ！

というわけで，つらつらっとバグ取りのコツを書いてきました。いかがでしたでしょうか？　なに，もの足りない!?　それではここで最大にして最終的な奥義を最後に伝授いたしましょう。それは，

最終奥義っ！　あ，バグないや，らっきーっ！　であります。ようするにバグさえなけりゃバグ取りなんてしなくていいわけです（あっこらっ！　石を投げるなっ！）。やっぱりちょっとずつ確実に作る！　これにかぎります。せっかくのX-BASICには関数なんていいものがあるんだからこいつを使わない手はありません。

めいっぱい小さい関数にプログラムを区切って１つひとつの関数を確実に作ることです。せっかくの特徴なんだから使わにゃそんそんです。

ま，でもなんだかんだいってもバグを出すのもそれはそれでまた楽しいもんですよ。そりゃ，バグを取るのはつらいし，それが１週間も２週間も続いたらディスプレイを金属バットでなぐってやろうとも思ってしまいますけどね。でも，いきなりカラフルな色で画面がいっぱいになったり，予想だにしなかった音がピロピロと鳴ったり……。エラーメッセージの出るバグでも「おお，こんなメッセージがあるのか」とかこっちの予想もしない事態が起こったり，思わず意表をついてくれることも多々あるんですよね（もっとも意表をつくからバグなんであって，原因がわかっていたら取っちゃうからね）。

そんな次第で「バグとハサミは使いよう」などと思ってしまう私でありました。皆さまもバグを楽しみながら精進していただきたいなと切に願う今日この頃でありました。じゃんじゃん。

（注1）マイクロソフト系BASIC＝マイクロソフトのCP/M用であったM-BASICが文法的な元になっているBASICのこと。おそらく世界中でいちばん普及した言語である。マイクロソフトの手掛けたN-BASIC（PC-8001）からF-BASIC386(FM-TOWNS)，マイクロソフトとは直接関係ないが文法的に近いN88BASIC86（PC-9801）やHuBASIC（X1など）なども含めていうことが多い。ちなみに同じくマイクロソフトから出ているBASICでQuickBASICというのもあるがこれは文法的にまったくいままでのマイクロソフトBASICと違うので普通は含まれない。

リスト6

```
 5 str strbuf
10 m_init()
20 m_alloc(1,1000):m_assign(1,1)
25 strbuf="l4@71o5h>e"
30 m_trk(1,strbuf)
40 m_play()
```

TRONを作る

なぜかX-BASICにはTRON（トレースオン)/TROFF命令がありません。これは実行中の行番号を画面に表示してくれるという機能を持った，一般のBASICではお馴染みのデバッグ用コマンドですが画面を壊したりするのであまり役に立たないというか，結構期待はずれに終わるコマンドとしても有名ですね。でも，ゴチャゴチャしたプログラムで，わけがわからなくなったときにはたまに役に立つこともあります（当然か）。

あまり使わないとはいっても，ないというのはちょっとしゃくですね。ということで，X-BASICで強引にTRON相当の処理をするプログラムを作ってみることにしましょう。

でも，いったいどういうふうにすればいいのでしょうか？　ここではデバッグ対象となるBASICプログラム全体を書き換えてみることにしました。

要するに，プログラムの各行の初めにその行番号を表示するprint文をつけてやればいいわけですね。案外簡単。で，そのとおりプログラムしたのが下のリストです。勝手にファイル名を決めるので，ファイル名制限いっぱいの長いファイル名やT_で始まるファイル名は使わないようにしてください。使い勝手を考えて同名ファイルのチェックや警告は行っていません。最悪の場合，既存のファイルを壊す可能性がありえます。

さて，これだけでもふつうのTRONと同じ効果はあるのですが，プログラムでやっているのですから，もっともっと賢く柔軟な処理をさせることも可能です（あまり長い行は使わないようにしようね）。リストの注釈部分のようにlocateを使えば表示位置を固定してやることも簡単です。さらに，locate直前にカーソル位置を読み出して行番号表示後に復帰するとか，1行ごとにキー入力待ちを入れるとか，指定範囲だけをトレースするとか，特定の変数を同時に表示してやるとか……。

ほら，なんとなく使えそうな気がしてきませんか。1990年代のデバッグ環境はやっぱりこうありたいですね。

```
 10 /* TRON.BAS
 20 int i,j,k
 30 str lin[255],nam,nlin[255],l,fun
 40 fun=" print "
 50 /*fun=" locate 0,0:print "
 60 input "ファイル名を入れてください";nam
 70 i=fopen(nam+".bas","r")
 80 j=fopen("T_"+nam+".bas","c")
 90 while freads(lin,i)<>-1
100 l=left$(lin,5)
110 nlin=l+fun+l+":"+right$(lin,len(lin)-5)
120 fwrites(nlin,j)
130 fputc(13,j):fputc(10,j)
140 print nlin
150 endwhile
160 fcloseall()
170 end
```

X-BASICでゲームを作る

豪華版SCRAMBLE

Kioi Makoto
紀尾井　誠

「なにを作ろうか？」「やっぱりゲーム」という人のためのX-BASIC入門です。3Dスクロール&ローリングはBASICで実現できるでしょうか。ではできるだけ簡単な処理でどこまでできるか？　をテーマにいってみましょう。

さて，今月もページが少ないんで飛ばしていくぞ。

まず，我々の前にあるのはBASICだ。変なBASICでサンプルを載せるなという意見ももっともらしく聞こえるが，C言語が使える人よりはBASICを使える人のほうがずっと多い。

プログラムを作る側からすれば，どんな言語を使うかというのは全然本質的な問題ではない。なにかの言語でプログラムを組める人ならほかの言語がまったく使えないということはありえない。

といっても，プログラミングの才能というものがあったり，特異な能力が要求されるわけではない。むしろプログラミング以外の能力が大切となる。プログラムする対象がわかっていなければプログラムなどできるはずがないのだ。自分が理解していることでなければプログラム化できない。これは当たり前。「ゼビウスみたいなゲームが作りたいんですが，スクロールがよくわかりません」とか「源平討魔伝のようなゲームを作ってるんですがデカキャラの動かし方がわかりません」「チェスゲームの思考ルーチンがわかりません」という人は歯を磨いて出直してきなさい。

誰にでも自分の得意分野とか，興味のあることがあるはずだ。まずはそれから始めるのがいい。あとは「プログラムを作るんだ」という気持ちだけ持っていれば自然とプログラムはできあがる。ではプログラミングに関する知識は？　まず，マニュアルを読みなさい。読みにくくても我慢して読む。わからなくてもとにかく読む。みんなそうしてきたのだから。

ゲーム画面

ゲームを作りたい

パソコンユーザーの多くはゲーム指向だ。FuzzyBASICコンパイラや初期のタイニイBASICのようにゲームを作るために言語を作った人も大勢いる。マシン語を覚えたのもゲームを作るため，という人も実は多い。X68000ユーザーはゲームばかりしているといわれたら胸を張って当然だと答えること。

できれば，ゲームをしているあいだもいろいろと考えてほしい。そこには技術的な限界への挑戦があるからだ。ラスタースクロールをしていたら水平帰線割り込みを思い出すとか，128個以上のスプライトを動かしていたらどうやってるんだろうとか，拡大縮小はメモリにパターンを持っているんだろうかとか。大変なことをしているなあと，BASICでは無縁な処理でも一応察しをつけておくとよい。

さて，プログラミングをするなら，ここはゲームしかない。ということで，いきなりX-BASICでゲームを作ることにする。ゲームの華といえばやはりシューティングゲーム。しかし，ピコピコした画面を作ってもBASICに失望するだけだから，ここではできるだけ簡単で思いっきり派手なピコピコゲームを目指す。

といっても，すぐに素晴らしいアイデアが湧いて出るはずないから，少し考える。「BASICではろくなものはできない」というのもおそらく真実だ。しかし，X68000というハードウェアの可能性を考えてみたとき，これは再考の余地がある。ハードは凄いんだから，思いっきりハードに依存したプログラムにすれば……。

ふと，以前，MZ-1500用のショートプログラムとして発表された「SCRAMBLE(作：熊谷聡)」というゲームが頭に浮かんだ。BASIC＋マシン語で十分面白いゲームにしあがっていたなあ。X68000のハードウェアを考えてみると……面白い。と，突然移植に走ることにする。

画面処理を考える

SCRAMBLEというゲームはフライトシミュレータ風のシューティングゲームだった。といっても，X68000ならMZ-1500と同じというわけにはいかない。ここで基本構想をまとめる。

1)　地面はタイルパターンで3D風にスクロールする（ex.スペースハリアー）
2)　画面は移動方向により，ローリングする（ex.アフターバーナー）
3)　基本的な部分は原作に同じ

さて，これをBASICで行うための手はひとつしかない。パレットを使う。よくある手だ，といってもわからない人にはわからないはずだから図1を見てほしい。

4×4の部分で構成されたタイルがある。

図1　パレットの使い方

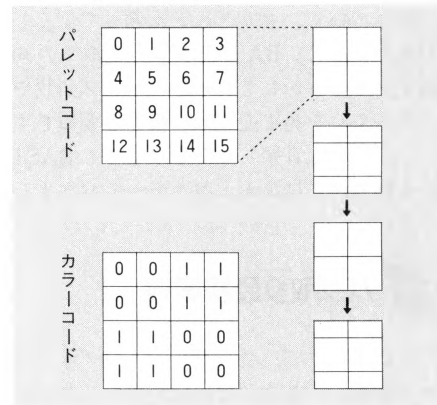
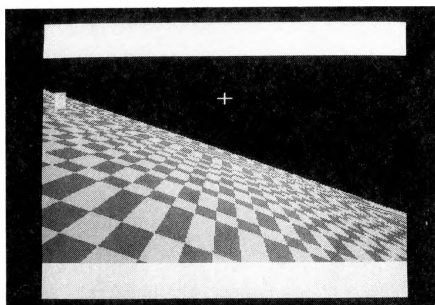
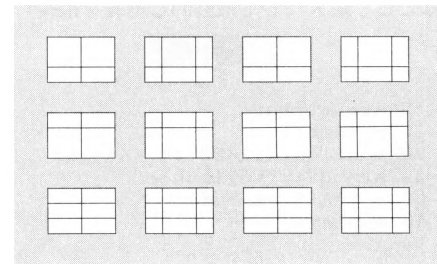



図2　12色での塗り分け

これを16色の違う色で塗り分ける。さらに，その色のパレットを白と黒のカラーコードに変えてみよう。そして色の割り当てを少しずつ変える。すると動いているように見える。色を塗り直すのではなく，色の割り当てを変えるだけなので高速だ。

この例では16色使っているが，16色モードで16色全部使ってしまうと地面以外の部分（空）までチカチカしてしまうので，しかたなく12色の塗り分けにした。この場合のパレットの変え方は図2のようになる。見てわかるとおり，横方向の1列目と3列目，2列目と4列目は色が反転しているだけなので，プログラムではそれも利用することにする。

この順番で地面を塗り分けて，このようにパレットを変えると前後左右に動いているように見えるはずだ。これを遠近感のついた地面に割りつければばいい。3Dスクロール部分はなんとかなりそうだ。

次は左右のロール，地面を傾ける処理を行う。これをリアルタイムで計算できるほどBASICは速くない。これも手はひとつ。画面切り換えだ。

X68000のグラフィック画面は16色モード時1024×1024の大きさを持っている。ゲームで使用する画面は256×256あれば十分。要するにscreen 0,0,1,1モードで16画面分の絵が描けるわけだ。これを目一杯使えば左右に7段階傾けることができる（真ん中を入れて15画面)。画面を多少マスクすれば，9段階ずつにすることもできるのだが，狭くなるのでやめておいた。

画面のどの部分を表示するかはhome( )関数で設定すればよい。

要するにあらかじめ全部描いておくわけだ。これらはいわゆるオフラインの処理だから，いくら時間がかかってもかまわない部分だ。思いっ切り遅くてもかまわない。

プログラムの実際

しかし，傾いたタイルパターンの地形を描くのはちょっと面倒そうだ。方法としては，

1)　レイトレーシングもどきを使う
2)　真っすぐなやつを描いて回転させる
3)　なんとなくそれっぽく描く

というのが考えられる。1)は処理が重そうだが，もっとも単純な計算ですむので意外と軽くてすむかもしれない。正確だし，平面1枚しかオブジェクトがないし……，シンプルで惹かれるものがある。2)はどう考えても重い。ここでは特に正確さを要求されないので，3)で安直にすませておく。

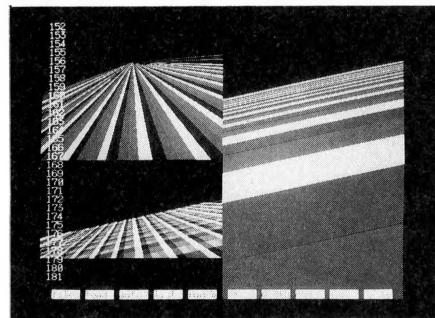
まず真っすぐなものを描いて，その描き方を回転させることにしよう。安直といっても，ある程度の数学は使うので各自，図を描いて考えてほしい。義務教育はこういうときに役立てよう。プログラムではのちのちの回転を考慮して「circle( )関数で直線を引く」という暴挙に出ている。これだと1点を必ず通る度単位（ラジアンではない）の直線が簡単に引ける。コンパイルするとライブラリの仕様の違いか，誤動作するので注意。

X-BASICには画面との論理演算つきのグラフィック関数がないので，縦，横2つの絵を描いてプログラムで合成する。そして，img_save( )したものを，ロード開始座標を指定してimg_load( )すれば楽勝……と思っていたら，伏兵がいた。

img_load( )ではなぜか512以上の座標にロードできない。さらに1024×1024のモードで全体をセーブするモードがない。これは1画面ずつロードして転送することで対処した。マニュアルの不備とかタコな仕様というのはわりとありがちなので，いちいちめくじらをたてない。変なら変なりに，バグがあればバグを避けてプログラムを作る。昔はもっと凄かったもんだ。

screen 0,0,1,1で初期化したあと描画し，img_scrn(1,3,1)として画面を壊さずに画面モードを変え，img_save( )で .gl 3 を指定してグラフィックRAM全体をセーブ。同様にロードして，img_scrn(0,0,1)で元の画面に戻す，という操作で一括ロードできることがあとでわかったが，プログラムでは行っていない。これにすると数行プログラムを短くできる。

これもimg_scrn( )という関数を知らなかったためだ。くそー，カラーイメージユニットのマニュアルにあった説明や初代の福袋に入っていたimage. fncにはこんな便利な関数はなかったぞ。教訓：マニュアルはよく読もう。

なお，都合により画面作成プログラムは右傾斜用と左傾斜用で異なる。ひとつ

地面を作る

にすることもできるがわざわざするのも面倒なので，リスト2の部分を打ち変えてもう一度実行し，最後に40行の，

for j=1 to 7

という部分を，

for j=0 to 0

に変えてさらに一度実行してほしい。これですべてのデータが揃う。

これでベースはできあがった。画面にデータをロードしパレット変更部分を作ってみると結構ちゃんと動く。画面がちらつくのはパレット切り換えや表示位置切り換えのIOCSが画面の垂直同期を見ていないためだ。アセンブラの使える人はちゃんとした外部関数を作ったほうがいい。さあ，あとはゲーム本体を作るだけだ。

本体は簡単，面倒な当たり判定は画面の1点で判断できる（だからこのゲームを選んだのだが)。ゲームは「画面上に現れる敵を捕捉し，単位時間にどれだけ撃墜できる

リスト1

```
  10 int y,cl
  20 float s,j2,y2,d
  30 screen 1,0,1,1
  40 for j=1 to 7
  50 d=4
  60 wipe():window(0,0,255,255):line(128,0,128,75,1)
  70 j2=pi()*j*d/180
  80 for i=0 to 121
  90   s=atan(((i-60)*10#)/100)*180/pi()+(j*d)
 100   circle(128,75,500,(i mod 4)+1,(-s-270) mod 360,(-s-270)
mod 360)
 110   paint(130,0,(i mod 4)+1)
 120 next
 130 y=128#*tan(j2)
 140 paint(129,0,0)
 150 window(256,0,511,511)
 160 y2=0
 170 for k=0 to 40
 180   y2=pow(y2,1.44#)/7+k/2
 190   line(256,75+y+cos(j2)*(y2+35),511,75-y+cos(j2)*(y2+35),(
k mod 3)+1)
 200   paint(511,511,(k mod 3)+1)
 210 next
 220 window(0,0,511,511):fill(0,256,255,511,0)
 230 for q=0   to 255
 240   print q
 250   for r=0   to  255
 260     c1=point(r+256,q)
 270     if c1<>0 then pset(r,q+256,(c1-1)*4+point(r,q))
 280   next
 290 next
 300 img_save("map"+str$(7-j)+".gs0",0,256)
 310 next
```

リスト2

```
 70 j2=-pi()*j*d/180
100 circle(128,75,500,3-(i mod 4)+1,(s-270),(s-270))
110 paint(0,0,3-(i mod 4)+1)
200 paint(256,511,(k mod 3)+1)
300 img_save("f:map"+str$(j+7)+".gs0",0,256)
```

▶今年こそは個人誌を出したいぞ(題名はもう決まっているんだけどな)。
丸藤　俊之（21）X68000ACE-HD　神奈川県

か」というものだから，時間も計らなきゃいけない。これにはいろいろ手があるが，ここではFM音源が演奏中かどうかで判断している。これならコンパイルしても瞬時に終わることはないし，タイマーの数値化も必要ない。

画面のスクロールもこれだけでは味気ないので，上下を少しマスクして水平線を動かすようにしてみた。そのほか，タイトなプログラムを愛する私としては，速度最優先でパレット切り換えを展開してあるのが少々心残りとなっている。

拡張の指針

ページと時間の余裕がなくて今回はここまでとする。このままでは最低限のゲームでしかないので，例によってあとは皆さんで拡張してほしい。

画面関係の解説ばかりになってしまったが当初の予定では FM 音源4パートのBGMと数種の効果音(FM音源)が入ることになっていた。ここで効果音の扱いについて少し触れておく。

まず，適当にミュージックトラックを確保する。まあ，

```
for i=1 to 8
    m_alloc(i,2000)
next
```

のような具合だ。これがメイン BGM 用となる。さらに，

```
for i=9 to 80
    m_alloc(i,20)
next
```

のように効果音用のトラックを確保する。見慣れないかもしれないが，ミュージックトラックは80本まで取ることができる。これに効果音を入れ，m_assign( )で切り換えるわけだ。

仕様として BGM は4声，効果音は最大3つ同時に鳴るとする。この場合のFM音源チャンネルの割り当ては，たとえば，

1　BGM（主旋律1）
2　BGM（主旋律2＆バッキング）
3　BGM（バッキング）
4　BGM（リズム）
5　効果音1
6　効果音2
7　効果音3
8　未使用

のようになる。

効果音1用のチャンネル5を見て演奏中ならチャンネル6を見て……のように空きチャンネルを探してその効果音を m_assign( )し，m_play( )すると適切な効果音が鳴るはずだ。

効果音数が少なければこのようなことは不要だが，私は個人的に効果音が多いほうが好きだ。昔，SFってのは絵だねぇといった人もいるが，最近ゲームってのは音だねぇと思うことが多い。BGM に加え効果音の比重が大きい。スペハリの敵の出現音とかR-TYPEの対空レーザー，グラディウスのパワーアップ，ポピュラスの沼……。うーん，効果音は多いほうがいいな。

ジェット機風のノイズもほしいから，これはチャンネル8に割り当てて，鳴らしっぱなしにし，ときどき効果音チャンネルから高度に従った音程になるような y コマンドを流し込んでやる。トラックはたくさんあるんだ。

いうまでもなく，OPMDを組み込めば自動的に MIDI 対応となるし，ゲーム時間を FM 音源で計っているのも実は BGM の終わりを1面の終わりにするためだったりする。

そのほか，ミサイルが情けない（あらかじめメモリスイッチで¥はバックスラッシュに変えること）とかキャラクター定義が凄く手抜きされているとか，改良すべき点はいくらでもあるぞ。では，健闘を祈る。

リスト3

```
 10 int x,y,dx,dc,dy,sco,ddx,u=18000          :/* initialize
 20 int st,ud,rl,vx,vy,tx,ty,tvx,tvy,miss
 30 char m(32767),defs(255)
 40 screen 0,0,1,1
 50 color 13:locate 0,15:print string$(32," ");
 60 locate 0,0:print string$(64," ")
 70 console 2,12,0:color 3
 80 tx=120:ty=100
 90 m_alloc(2,100):m_assign(2,2):m_trk(2,"|:20r1:|")
100 window(0,0,1023,1023):sp_init():sp_disp(1)
110                                            /* preparate
120 symbol(0,0,"●",1,1,0,4,0)
130 for i=0 to 255:defs(i)=point(i mod 16,i¥16):next
140 sp_def(0,defs)
150 for i=1 to 14
160 x=(i mod 4)*256 :y=(i¥4)*256
170 img_load("map"+str$(i)+".gs0")
180 get(0,0,255,255,m)
190 put(x,y,x+255,y+255,m)
200 next
210 img_load("map0.gs0")
220 locate 15,4:print"+"
230 m_play(2)
240                                            /* main
250 while m_stat(2)=1
260 if miss=0 then sp_disp(1)
270 st=stick(1):ud=((st-1)¥3)-1
280 if st <> 0 then rl=((st-1) mod 3)-1 else rl=0
290 if strig(1)=1 then shoot()
300 locate 0,3:print sco
310 if abs(vx)<29 then vx=vx+rl*3
320 vy=vy+ud*2:if abs(vy)>14 then vy=vy-ud*2
330 if vx<0 then dx=(dx+1)mod 4
340 if vx>0 then dx=(dx+3)mod 4
350 ddx=vx¥4
360 if miss=0 then {
370 if int(rnd()*20)=1 then tvx=(rnd()*2)*4-2:tvy=rnd()*3-10
380 tx=tx+tvx-vx:ty=ty+tvy-vy
390 if tx>0 and tx<768 and ty>0 and ty<736 then sp_move(0,tx¥3,
ty¥3,0) else miss=30
400 sp_disp(0)} else edisp()
410 vx=sgn(vx)*(abs(vx)-1)
420 home(0,(((ddx+7) mod 4)*256+1024)mod 1024,(((ddx+7) ¥4)*256
+1024+vy)mod 1024)
430 dy=(dy+1) mod 3:if dx¥2=0 then dc=17510 else dc=-17510
440 if (dx mod 2 =0) then {
450 if miss=0 then sp_disp(1)
460                                            /* palette
470 switch dy
480 case 0
490  palet(1,u-dc):palet(2,u-dc):palet(3,u+dc):palet(4,u+dc)
500  palet(5,u-dc):palet(6,u-dc):palet(7,u+dc):palet(8,u+dc)
510  palet(9,u+dc):palet(10,u+dc):palet(11,u-dc):palet(12,u-dc)
520 sp_disp(0):break
530 case 1
540  palet(1,u+dc):palet(2,u+dc):palet(3,u-dc):palet(4,u-dc)
550  palet(5,u-dc):palet(6,u-dc):palet(7,u+dc):palet(8,u+dc)
560  palet(9,u-dc):palet(10,u-dc):palet(11,u+dc):palet(12,u+dc)
570 sp_disp(0):break
580 case 2
590  palet(1,u-dc):palet(2,u-dc):palet(3,u+dc):palet(4,u+dc)
600  palet(5,u+dc):palet(6,u+dc):palet(7,u-dc):palet(8,u-dc)
610  palet(9,u-dc):palet(10,u-dc):palet(11,u+dc):palet(12,u+dc)
620 sp_disp(0):endswitch
630 } else { switch dy
640 case 0
650  palet(1,u+dc):palet(2,u-dc):palet(3,u-dc): palet(4,u+dc)
660  palet(5,u+dc):palet(6,u-dc):palet(7,u-dc): palet(8,u+dc)
670  palet(9,u-dc):palet(10,u+dc):palet(11,u+dc):palet(12,u-dc)
680 sp_disp(0):break
690 case 1
700  palet(1,u-dc):palet(2,u+dc):palet(3,u+dc):palet(4,u-dc)
710  palet(5,u+dc):palet(6,u-dc):palet(7,u-dc):palet(8,u+dc)
720  palet(9,u+dc):palet(10,u-dc):palet(11,u-dc):palet(12,u+dc)
730 sp_disp(0):break
740 case 2
750 palet(1,u+dc):palet(2,u-dc):palet(3,u-dc):palet(4,u+dc)
760 palet(5,u-dc):palet(6,u+dc):palet(7,u+dc):palet(8,u-dc)
770 palet(9,u+dc):palet(10,u-dc):palet(11,u-dc):palet(12,u+dc)
780 sp_disp(0):endswitch}
790 endwhile
800 end
810                                            /* sub_function
820 func edisp()
830 miss=miss-1
840 if miss=0 then tx=rnd()*600+28:ty=rnd()*450+30
850 sp_move(0,tx¥3,ty¥3,0)
860 endfunc
870 func shoot()
880 color 6
890 for i=1 to 7
900 locate i*2, 12-i:print"/":locate 31-(i*2), 12-i:print"¥"
910 next
920 color 3:cls:locate 15,4:print"+"
930 if (tx+8)¥24=15 and (ty+8)¥48=4 then beep:miss=10:sco=sco+1
:tx=0
940 endfunc
```


▶ X68000の次機は，みなさん X68020か X68030だと思っているようですが，これは名前を省略しづらい。それなら単に，X68000 IIのほうが良いのではないだろうか（省略時には X68 IIとなる）。
松井　達也（26）X68000ACE，MZ-2500　神奈川県

CARD.FNCを活用する

カードゲームを作ろう

Mounai Toshiyuki
毛内 俊行

カードゲームが簡単にできるCARD.FNCを使ってトランプゲームを作ってみましょう。サンプルは「99」，ちょっとマイナーなトランプゲームです。では，カードゲームに必要な処理の基本的な考え方を中心におとどけします。

カードゲームは特に高速性を要求されることもなくBASIC向きの題材だといえますね。しかも一度基本的な部分を作ってしまえば，あとはルールの違いをプログラムするだけでゲームができあがります。

今回はサンプルに用意した99ゲームを使ってCARD.FNC（65ページ参照）でゲームを作るときのノウハウを説明します。

プログラムの設計

さっそく，プログラムを作っていきましょう。ここでは実際にプログラムを作るときの手順を踏んで，どのようにプログラムを作ったらいいのかを考えてみましょう。

1)　メインプログラムの作成

プログラムの中心的存在であるメインプログラム。やはり，これを作ることから始めなくてはいけません。ここでは，あまり複雑な作業は行いません。まず，全体のプログラムの大まかな流れを作ってやるのです。たとえばゲームプログラムなら，

1. 初期設定
2. ゲーム実行
3. ゲームオーバー処理

といった順序で進行するでしょう。つまりメインプログラムでは，これらの項目を関数で記述してやり，順番に実行してやればいいのです。そうするとメインプログラムの内容は，およそこんな感じになります。

```
while  1
   INIT ( )
   GAME ( )
   GAMEOVER ( )
endwhile
end
```

このように各項目を関数にしてしまえば，プログラムの流れがつかみやすくなります。このプログラムのwhileループは無限ループなので，endwhileの直後にあるendが実際に実行されることはありません。しかし，なにかの拍子で実行がループを抜け出したりしたときに，エラー発生の歯止めとなりますし，なによりもメインプログラムが完結していないと，かゆいところに手が届かないみたいで落ち着かないので，一応「メインプログラムはここまで」という目印のつもりでつけています。

また，実際のリストでは初期設定の関数をプログラムの実行直後に1回実行すればいいものと，ゲームが始まるたびに実行しなければならないものの2つに分けています。この初期設定についてはあとでもっと詳しく説明していきます。

とりあえずこれで，ゲームの骨格をなすメインプログラムを用意できました。えっ？本当にこんなに短くていいのかって？

そうです。プログラムはなるべく短くするのが基本です。特にX-BASICは，関数の定義が自由自在なのですから，小さな関数プログラムをたくさん作って，本体のプログラムからそれらを呼び出してあげるのがコツなのです。

2)　初期設定関数の作成

どんなプログラムでも，必ず最初に実行するのがイニシャライズプログラム，すなわち初期設定です。実際のリスト中では，260行からのINIT，500行からのCINITが，それぞれ初期設定関数です。

この2つの関数の違いですが，INITがゲーム画面などの作成を主に行っているのに対し，CINITではプログラム中で使われる変数内容の初期化を行っています。とにかく，ここでゲームの準備をするのです。

3)　カードゲームに共通な関数を作る

一概にカードゲームといっても，その種類は星の数ほどあります。しかし，ゲームのルールは異なっても同じカードゲームなのですから，それぞれ共通の部分が存在します。たとえば次に紹介するシャッフルは，すべてのカードゲームで行われる作業です。さあ，それでは実際にそれらのプログラムを作成する方法を考えてみましょう。

●シャッフルする

シャッフルとはカードを切りまぜすることです。それでは，シャッフルするにはいったいどうやったらいいのでしょうか？

シャッフルされたカードには2つの特徴があります。それは，

　カードの順番が整っていないこと
　同じカードが2つ以上ないこと

この2つです。最初の条件だけだと，ただ適当に乱数でカードを決めてしまえばいいと思うのですが，それでは同じカードがダブって登録される可能性があり，2つ目の条件をクリアできません。それではまずいので，なにか新しい方法を考えましょう。

たとえば，乱数で発生させた数が過去に登録されていないかを，1回1回チェックするという方法があります。しかしこれは，カードが多くなるにしたがって，実行速度がガタっと落ちてしまいます。そこで別の方法を考えてみました。

まず，最初に52枚のカードを用意し，その中から任意にカードを2枚取り出します。このカードをそれぞれA，Bとします。いま引いたAのカードをBのカードの入っていた場所に，BをAの入っていた場所に置けば，この2つのカードのあいだでシャッフルが行われたことになります。

これを繰り返せば，それほど時間がかからずにシャッフルができるはずです。

まず，カードを52枚用意するところから作ります。これは簡単です。まず52枚のカードを格納する配列変数を用意します。ここではC1という配列を使うことにします。すると，次のようになります。

```
for  I=0  to  51
  C1(I)=I+1
```

next

配列C1にカードの内容を示すカード番号が登録されました。代入のところでI＋1としているのは使われるカード番号が0～51ではなく，1～52だからです。さて，カードの用意がすんだところで，シャッフルの本体のプログラムです。まず2枚のカードの交換の部分はこんな感じになります。

```
A＝rnd（）＊52
B＝rnd（）＊52
C＝C1（A）
C1（A）＝C1（B）
C1（B）＝C
```

最初の2行でカードを選んで，残りの3行で交換しています。本当はswap命令があると楽なんですが，どうやらX-BASICにはないみたいなので，入れ替え作業に変数Cを介しています。あとはこの作業を何回も繰り返せばいいのです。経験から，だいたいデータの数だけ繰り返せば綺麗にシャッフルされるようです。つまり，52枚のカードをシャッフルするのにおよそ50回実行すればいいわけです。ただしリストの中では倍の100回ほど実行しています。

また，rnd（）＊52となっているところを，rnd（）＊NとしておけばNの値を変えるだけで，シャッフルする枚数を変えることができます。Nは関数のパラメータとしておけば便利です。リストの590～670行が，シャッフルの関数になっています。

●カードを表示する

次にカードを表示する関数を作ってみましょう。「あれ？　カードを表示するなら，c_put命令があったじゃないか！」確かに，CARD.FNCにはカードを表示する命令として，c_putという命令が用意されています。しかし，この命令では実際にゲームを作るときに，どうしても不都合が生じてしまうのです。たとえば，

```
c_put（X,Y,N）
```

で，画面に表示したカードの右隣に，カードを表示しようとしたら，

```
c_put（X＋48,Y,N）
```

と，実行しなければいけないのです。確かに，グラフィック画面のどこにでも表示できるというのは便利ではあるのですが，実際にプログラムを作る側にとっては，かえって面倒な処理を強いられてしまいます。

やはり，隣にカードを表示するときには，X＋1と表現したくなるのが人情です。そこで，それらの作業を一手に引き受けてくれる関数を作ってしまいましょう。どうすればいいのかというと，カードの大きさにあわせて座標の変換を行ってからカードを表示する関数を作ってあげればいいのです。

カードの大きさは48×96ドットなので，X軸方向のパラメータに48を，Y軸方向のパラメータに96を掛けてから，c_put命令を実行してやります。実際には，

```
func　CDPUT（X,Y,N）
c_put（X＊48，Y＊96，N）
endfunc
```

と，こんな感じになります。この関数では，カードを表示するのに座標とカードが1対1で対応しているので，変な計算はいりません。この関数を使えばグラフィック画面が512×512ドット時に10×5枚のカードの表示ができます。リストでは690～770行がこの関数です。リストを見ると実際は，画面の左端にプレイヤーの名前を表示するスペースを空けるため，X軸方向は座標変換を行ったあとでさらに16を加えています。

それからもうひとつ，リスト中ではプレイヤーが人間かコンピュータかを調べて，カードを伏せて表示するか開いて表示するかを判定しています。人間がプレイヤーの場合はY軸のパラメータが必ず4になっているので，これを調べればいいわけです。

ただし例外として，カード番号のパラメータに100を加算してこの関数を実行すると，プレイヤーがコンピュータでもカードを開いて表示するように書かれています。この機能はゲームオーバーのときに全員のカードを開くのに使っています。このようにすでに用意されている関数だって，自分の使いやすいように作り変えると，プログラムを作成する速度やバグの発生率が，大きく変わってきます。道具は自分で使いやすいように使ってやるのがコツなのです。

●効果音

効果音というのは，カードゲームに限ったものではありませんが，やはり関数として作っておくと便利なものですので，紹介しておきましょう。効果音を発生させるのはリストの3330行からのOTO関数です。たとえば，これは一度ゲームを実行してからでなくては使えませんが，ゲーム中にブレイクキーを押して実行を中断して（グラフィック画面が残ったままでは見にくいので，一度CTRL-Dかwidth命令を実行したほうがいいでしょう）キーボードから，

```
OTO(PICO)
```

と実行してみてください。スピーカーから「ピコピコ」と音が聞こえるはずです。

リストを見てもらえばわかるのですが，この関数はFM音源の1トラックを使って，パラメータで渡された楽譜を演奏する関数です。つまりここで使われたパラメータPICOは，「ピコピコ」という音を鳴らすための楽譜を代入した文字変数だったのです。試しにキーボードから，

```
print PICO
```

と実行してみましょう。画面に楽譜データが表示されるはずです。この関数は，

```
func OTO(M;str)
m_init（）
m_trk（1，M）
m_play（）
endfunc
```

と，いった感じで作ればいいでしょう。ここでは1トラックしか使っていませんが，これを応用すれば複数のトラックを使った効果音を鳴らすことも簡単にできます。

なお，効果音はデフォルトの音源データを使っているので，音源データをいじったあとでは音が変わってしまうかもしれません。X68000のFM音源は，最大200種類も音色を定義できるのですから，なるべく元の音は壊さないようにしましょう。

ゲームをプレイする人に

さて，ゲームを実際にプレイする人にいくつか注意しておくことがあります。まず，先ほども少し触れましたが，このプログラムを実行するには今月発表されたCARD.FNCが必要です。まず，先にそちらを入力してください。それからプログラムを実行するとすぐに，役札の説明が必要かどうかを尋ねてきます。ここでYを入力するとゲーム画面に役札の一覧が表示されます。本当はこんなことを尋ねなくて，ずっと表示したままでもいいのですが，個人的にこの表示がどうしても気にいらなかったので表示しないようにできることにしました。

あと，表示速度が速いとか遅いとか思う人があったら，110行の変数TMRの値を変えてください。値を小さくするほど実行速度が速くなります（通常は500）。

＊　＊　＊

さて，今回は「カードゲームの作り方」みたいな感じで，いろいろといいたいことを述べてきましたが，結局ゲーム本体の作り方にはなにも触れませんでした。もっとも，ゲーム本体のプログラムというのはそれほど凝った技術は必要ないので（ものすごく凝った思考プログラムが必要な場合もあるけど）ちょっとBASICを触ったことがある人なら，簡単に作ることができるでしょう。

実際，カードパターンとカードを表示するプログラムを作らなくていいというのは，

▶生協の本棚でOh！Xを見たとき，一瞬「サイエンス」か見間違えた。
萩原　豊隆（20）X1G，MZ-700　長野県

カードゲームを作る人にとってはとても嬉しいことです。おそらく開発時間の半分はこのカードデータの作成に使われるのですから。しかも，カードが綺麗なのだから文句のつけようがありません（ここまで自画自賛するとあつかましい？）。

まったくの余談ですが，私は個人的にスペードのクイーンが好きです。彼女ってなんとなく美人じゃありませんか？（こりゃ失礼）

今回紹介した99というゲームは，私の周りでは結構ポピュラーなゲームだったのですが，本屋さんで「○×のトランプゲーム」みたいな本を10冊くらい見てもひとつも載っていないのでびっくりしてしまいました。一応，ルールは囲みのほうに紹介しておきましたので，ルールを知らない人はそちらをご覧ください。

99のルール

使うカードはジョーカーを除いた52枚。プレイヤーは2～5人が適当でしょう。まず，各プレイヤーに4枚ずつカードを配り，残ったカードを山札としてテーブルの中央に積んでおきます。プレイヤーが順々にカードを1枚ずつ場に捨てていき，1枚捨てるたびに新しいカードを1枚，山札から引きます。場に捨てられたカードはカードの数がどんどん加算されていきます（J～Kの絵札はすべて10とする）。カードを捨てたとき，場に加算された数が99を越えたら，その人の負けになります。

カードのなかには役札という特殊な役割を持った札があります。この役札を使って，プレイヤーは場の数やプレイヤーの順番を，コントロールすることができます。たとえば場の数が95のときにJを捨てると，本来なら場の数は105になってしまい，その人の負けになりそうです。しかし実際は，場の数が90以上のときにJはスキップという役札になるので，場の数は95のまま変わらず，次のプレイヤーをひとり飛ばしてゲームが続行されるのです。

なお，このルールは日本大学コントラクトブリッジクラブで一般に使われているルールです。このゲームはマイナーなわりに，ローカルルールが結構多いので「あれ？　ルールが違うぞ」という方は勘弁してくださいね。

表1　役札の説明

いつでも使える役札（常に役札である札）	
A	（ハート）　場の数を0にする
A	（ダイヤ）　場の数を99にする
場の数が90以上で役札になる札 （通常はふつうの札として使われる）	
A	プレイヤーの指名 （ハート，ダイヤを除く）
4	プレイヤーの順番が逆回りになる
9	場の数を99にする
10	場の数から10を引く
J	プレイヤーをひとり飛ばす

リスト1

```
  10 dim int C1(51),C2(31),PL(19)
  20 dim int YK(51)={                                 /*役札内容
  30 /*-A--2--3--4--5--6--7--8--9-10--J--Q--K------------
  40   95, 0, 0,92, 0, 0, 0, 0,93,94,91, 0, 0, /*スペード
  50    6, 0, 0,92, 0, 0, 0, 0,93,94,91, 0, 0, /*ハート
  60    3, 0, 0,92, 0, 0, 0, 0,93,94,91, 0, 0, /*ダイヤ
  70   95, 0, 0,92, 0, 0, 0, 0,93,94,91, 0, 0  /*クラブ
  80 }
  90 dim str NM(4)={"うさぎさん","かめさん",
 100          "ぶたさん","ねこさん","  あなた"}
 110 int PN,PV=1,BA=0,BP,TMR=500,CX=0,YS=0
 120 str PICO="@56o5v15l20ECEC",TON="@59o3V15l4C"
 130 str PARA="@64o3v15l1c"
 140 /*
 150 /*--- MAIN PROGRAM ---
 160 INIT()
 170 while 1
 180   cls:wipe()
 190   CINIT()
 200   GAME()
 210   GOVER()
 220 endwhile
 230 end
 240 /*--------------------
 250 /*
 260 func INIT()
 270 str A,TT="Ninety-Nine"
 280 cls
 290 repeat
 300   input"役札の解説が必要ですか?(y/n)",A
 310   if A="Y" or A="y" then YS=1:break
 320 until A="N" or A="n"
 330 m_alloc(1,200):m_assign(1,1)
 340 randomize(val(mid$(time$,4,2)+right$(time$,2)))
 350 screen 1,1,1,1:palet(1,0):console ,,0:locate ,,0
 360 apage(3)
 370 fill(0,0,511,511,8)
 380 symbol(231,9,TT,1,2,2,0,0):symbol(230,8,TT,1,2,2,15,0)
 390 box(390,141,500,336,0):box(389,140,499,335,15)
 400 apage(1)
 410 for I=0 to 4
 420   for J=0 to len(NM(I))/2
 430     K=I*96+J*16+16
 440     symbol(0,K,(mid$(NM(I),J*2+1,2)),1,1,1,15,0)
 450   next
 460 next
 470 apage(0)
 480 endfunc
 490 /*
 500 func CINIT()                     /* カード初期化
 510 int I
 520 for I=0 to 51:C1(I)=I+1:next
 530 for I=0 to 31:C2(I)=0:next
 540 for I=0 to 19:PL(I)=0:next
 550 SFL(52):DEALALL():BAPRT()
 560 if YS=1 then YSETU()
 570 endfunc
 580 /*
 590 func SFL(N)                      /* シャッフル
 600 int A,B,C,I
 610 MSG(30,12,"シャッフル中です")
 620 for I=0 to 99
 630   A=rnd()*N:B=rnd()*N
 640   C=C1(A):C1(A)=C1(B):C1(B)=C
 650 next
 660 MSG(30,12,chr$(5))
 670 endfunc
 680 /*
 690 func CDPUT(X,Y,N)                /* 1枚カード表示
 700 if N=0 then {
 710   c_put(X*48+16,Y*96+8,59)
 720 } else {
 730   if N<100 and Y<4 then N=0
 740   if N>=100 then N=N-100:if N=0 then N=59
 750   c_put(X*48+16,Y*96+8,N)
 760 }
 770 endfunc
 780 /*
 790 func DEALALL()                   /* 画面にカードを並べる
 800 int I,X,Y
 810 for I=0 to 3
 820   PL(I)   =C1(I+32):C1(I+32)=0
 830   PL(I+4) =C1(I+36):C1(I+36)=0
 840   PL(I+8) =C1(I+40):C1(I+40)=0
 850   PL(I+12)=C1(I+44):C1(I+44)=0
 860   PL(I+16)=C1(I+48):C1(I+48)=0
 870 next
 880 for Y=0 to 4
 890   for X=0 to 3
 900     OTO(TON)
 910     CDPUT(X,Y,PL(X+Y*4))
 920   next
 930 next
 940 X=rnd()*10+10
 950 for Y=0 to X
 960   PN=Y mod 5:NM_MARK(PN)
 970   for I=0 to TMR/8:next
 980 next
 990 endfunc
1000 /*
1010 func Y_TEST(N)                   /* 役札のチェック
1020 int A,B
1030 A=YK(N-1)/10*10:B=YK(N-1) mod 10
1040 if A>BA then return(0) else return(B)
1050 endfunc
1060 /*
1070 func NM_MARK(N)                  /* 名前のマーク表示
1080 apage(2)
1090 fill(0,0,14,511,0)
1100 fill(0,N*96+11,14,N*96+100,4)
1110 apage(0)
1120 endfunc
1130 /*
1140 func KY_GET()                    /* 数字キー入力
1150 str KY
1160 while inkey$(0)<>""
1170 endwhile
1180 repeat
1190   KY=inkey$
1200 until asc(KY)>47 and asc(KY)<57
1210 return(val(KY))
1220 endfunc
1230 /*
1240 func CD_SEL()                    /* カード選択(人間用)
1250 int A,X
1260 X=CX
1270 fill(X*48+16,490,X*48+64,493,6)
1280 repeat
1290   A=KY_GET()
1300   fill(16,490,208,493,0)
1310   if A=4 then X=X-1:if X<0 then X=X+4
1320   if A=6 then X=X+1:if X>3 then X=X-4
1330   fill(X*48+16,490,X*48+64,493,6)
1340 until A=5 or A=0
1350 fill(16,490,208,493,0)
1360 CX=X
```

▶編集室にもF-QIXがあるんですか。あれはいいですねー。あっいやいや，なんというか256色モードを使ったりという工夫がしてあるし……，まっ絵がうまいと言うのが一番でしょうが。
小林　到（19）X68000ACE-HD　長野県

```
1370 return(X)
1380 endfunc
1390 /*
1400 func PL_SEL()                  /* プレイヤー指名(キー入力部)
1410 int A,Y
1420 MSG(28,12,"プレイヤーを2,8キー")
1430 MSG(28,13,"で選択してください")
1440 MSG(28,14,"決定は5キーです")
1450 Y=PN
1460 repeat
1470   A=KY_GET()
1480   if A=8 then {
1490     Y=Y-1
1500     if Y<0 then Y=Y+5
1510   }
1520   if A=2 then {
1530     Y=Y+1
1540     if Y>4 then Y=Y-5
1550   }
1560   NM_MARK(Y)
1570 until A=5 or A=0
1580 MSG(28,12,chr$(5)):MSG(28,13,chr$(5)):MSG(28,14,chr$(5))
1590 return(Y)
1600 endfunc
1610 /*
1620 func GAME()                    /* ゲームの実行
1630 int A
1640 NM_MARK(PN)
1650 while 1
1660   BP=1
1670   if PN=4 then A=MAN_PLAY() else A=COM_PLAY()
1680   if A=1 then break
1690   PNINC()
1700 endwhile
1710 endfunc
1720 /*
1730 func PNINC()                   /* 次のプレイヤーへの処理
1740 int I,J
1750 for I=1 to BP
1760   PN=PN+PV
1770   if PN>4 then PN=PN-5
1780   if PN<0 then PN=PN+5
1790   NM_MARK(PN)
1800   for J=0 to TMR*3:next
1810 next
1820 endfunc
1830 /*
1840 func MAN_PLAY()                /* プレイヤー(人間)処理
1850 int A
1860 A=CD_SEL()
1870 return(DO_CD(A))
1880 endfunc
1890 /*
1900 func COM_PLAY()                /* プレイヤー(COMPUTER)処理
1910 int A,I
1920 for I=0 to TMR*5:next
1930 A=CD_SELCOM(PN)
1940 return(DO_CD(A))
1950 endfunc
1960 /*
1970 func CD_SELCOM(N)              /* カード選択(コンピュータ用)
1980 int A,B,I
1990 for I=0 to 3
2000   A=PL(I+4*PN)
2010   if BA<90 and YK(A-1)<90 then break
2020   if BA>=90 and Y_SIM(A)<100 then break
2030 next
2040 if I=4 then return(rnd()*4) else return(I)
2050 endfunc
2060 /*
2070 func DO_CD(N)                  /* カード操作
2080 int C
2090 C=PL(N+4*PN):PL(N+4*PN)=0
2100 SUTE(C,N)
2110 YAKU(C)
2120 if BA<100 then PL(N+4*PN)=DROW(N)
2130 if BA<100 then return(0) else return(1)
2140 endfunc
2150 /*
2160 func SUTE(N,A)                 /* カードを場に捨てる
2170 int X,Y
2180 CDPUT(A,PN,0)
2190 X=rnd()*40+400:Y=rnd()*80+151
2200 c_put(X,Y,N)
2210 OTO(TON)
2220 C2SET(N)
2230 endfunc
2240 /*
2250 func C2SET(N)                  /* 場に捨てたカードの記憶
2260 int I
2270 for I=0 to 30
2280   if C2(I)=0 then break
2290 next
2300 C2(I)=N
2310 endfunc
2320 /*
2330 func BAPLS(N)                  /* 場の数の加算処理
2340 N=N mod 13
2350 if N=0 then N=13
2360 if N>10 then N=10
2370 BA=BA+N
2380 BAPRT()
2390 endfunc
2400 /*
2410 func DROW(X)                   /* 新しいカードを引く
2420 int A,I
2430 while 1
2440   for I=0 to 31
2450     if C1(I)<>0 then {
2460       A=C1(I):C1(I)=0
2470       CDPUT(X,PN,A)
2480       return(A)
2490     }
2500   next
2510   fill(390,141,499,335,0)
2520   for I=0 to 31
2530     C1(I)=C2(I):C2(I)=0
2540   next
2550 SFL(32)
2560 endwhile
2570 endfunc
2580 /*
2590 func YAKU(N)                   /* 場に捨てた札の処理
2600 int A
2610 A=Y_TEST(N)
2620 switch A
2630   case 1 :BP=2    :YPRT("==== SKIP ==== ",PICO):break
2640   case 2 :PV=PV*-1:YPRT("=== REVERCE ===",PICO):break
2650   case 3 :BA=99   :YPRT("==== 99 ==== ",PICO):break
2660   case 4 :BA=BA-10:YPRT("===== -10 =====",PICO):break
2670   case 5 :SIMEI() :YPRT("==== 指名 ==== ",PICO):break
2680   case 6 :BA=0    :YPRT("==== ゼロ ==== ",PICO):break
2690   default:BAPLS(N):YPRT(chr$(5),"")          :break
2700 endswitch
2710 BAPRT()
2720 endfunc
2730 /*
2740 func BAPRT()                   /* 場の数の表示
2750 if BA<100 then {
2760   if BA<90 then color 3 else color 1
2770   MSG(30,10,"場の数="+right$(" "+str$(BA),2))
2780   color 3
2790 }
2800 endfunc
2810 /*
2820 func YPRT(M;str,S;str)     /* 役札の種類の表示
2830 MSG(30,12,chr$(5)):MSG(30,12,M)
2840 OTO(S)
2850 endfunc
2860 /*
2870 func Y_SIM(N)                  /* 役札の内容を調べる
2880 int A
2890 A=Y_TEST(N)
2900 N=N mod 13:if N=0 or N>10 then N=10
2910 switch A
2920   case 0 :return(BA+N) :break
2930   case 3 :return(99)   :break
2940   case 4 :return(BA-10):break
2950   case 6 :return(0)    :break
2960   default:return(BA)   :break
2970 endswitch
2980 endfunc
2990 /*
3000 func SIMEI()                   /* 次のプレイヤーの指名
3010 int A,I
3020 if PN<4 then {
3030   repeat:A=rnd()*5: until A<>PN
3040 } else {
3050   A=PL_SEL()
3060 }
3070 BP=0:I=PN
3080 repeat
3090   I=I+PV:BP=BP+1
3100   if I<0 then I=I+5
3110   if I>4 then I=I-5
3120 until I=A
3130 endfunc
3140 /*
3150 func GOVER()                   /* ゲームオーバー処理
3160 int X,Y
3170 str KY
3180 cls
3190 MSG(36,23,"GAME OVER!!")
3200 MSG(36,25,NM(PN)+"の負けです")
3210 MSG(36,28,"何かキーを押して下さい")
3220 OTO(PARA)
3230 for Y=0 to 4
3240   for X=0 to 3
3250     CDPUT(X,Y,PL(X+Y*4)+100)
3260   next
3270 next
3280 while inkey$(0)<>""
3290 endwhile
3300 KY=inkey$:BA=0:PV=1
3310 endfunc
3320 /*
3330 func OTO(M;str)                /* 効果音発生
3340 m_init():m_trk(1,M):m_play()
3350 endfunc
3360 /*
3370 func MSG(X,Y,M;str)            /* メッセージ表示
3380 locate X,Y:print M
3390 endfunc
3400 /*
3410 func YSETU()                   /* 役札説明
3420 color 1:MSG(28,21,"いつでも使える役札")
3430 color 3:MSG(28,22,"A : (ハート:場=0 / ダイヤ:場=99)")
3440 color 1:MSG(28,24,"場が90以上で使える役札")
3450 color 3:MSG(28,25,"A : プレイヤー指名(ハート、ダイヤは除く)")
3460 MSG(28,26,"4 : REVERCE")
3470 MSG(28,27,"9 : 場=99")
3480 MSG(27,28,"10 : 場-10")
3490 MSG(28,29,"J : SKIP")
3500 endfunc
```


▶家に届いたOh！Xを見て，一瞬たじろいでしまった。「なんで，LOGINがきてるんだろう？」
小倉　宏明 (22) X68000，X1/turbo　新潟県

Z80's BAR番外編

通信によるファイル転送

クロスケーブルをつないでマシンからマシンにデータ転送。RS-232Cの制御はX-BASICではできないと思い込んでいる人も多いのでは？　ここではX1turboとX68000をつないでプログラム転送に挑戦してみましょう。

Nishikawa zenji
西川 善司

マスター（以下M）：OK.I see.Good bye. See you.〈ガチャ〉

善司（以下善）：ねーねー，誰と電話してたの？

M：あー，最近影の薄いメアリーからだよ。春休みだからカナダへ帰ってるんだ。

長老（以下老）：もう5月じゃぞ。ふつう春休みなんかとっくに終わっとるぞ。

善：あぁ，それはきっと時差のせいだよ。

一同：はははは，そうか。そうか。

善：ところで電話ってやつは便利だね。あんな海の向こうの国の人間とも話ができるんだからね。

M：パソコンにも通信がありますがあれは電話代がねー。カナダじゃ市内通話はタダ，なんてところもあるそうですよ。

老：いやいや，電話を使わないパソコン通信だってあるぞ。パソコン同士をじかにつないで行うものじゃ。

善：は？　いったいどんなメリットがあるの，それ？

老：たとえばX1のファイルをX68000に転送したり，X68000のファイルをX1へ転送したり。

善：へぇー。そりゃ便利そうだな。僕なんかよく昔X1で作った音楽プログラムなんかをX68000で作り直したりするんだけど，一度X1でプリンタで打ち出したのを打ち込み直していたんだ。

M：音楽プログラムなんかはプログラムの大半がMML（ミュージックマクロランゲージ）ですから確かにファイルを持ってこられれば移植は楽ですよねぇ。

善：でも，機材が必要なんでしょ，どうせ。

老：んや。RS-232CクロスケーブルとBASICがあればできるんじゃ。

M&善：えぇーっ！

M：そのケーブルはいくらぐらいなんですか？

老：そうじゃな，3000円から7000円くらいじゃろう。ただし買うとき「モデム用」のを買ってはいけないぞ。必ず「クロスケーブル」というのを買わなければいかん。

善：BASICでほんとにそんなパソコン通信プログラムが書けるの？

老：X1turboに付属している「turboBASIC（ボーランドじゃないぞ）」とX68000付属の「X-BASIC」でちゃんと書ける！

通信パラメータとは

M：turboBASICにそんなことができる命令がありましたかね？

善：あ，ちょっと待てよ。たしか
　OPEN"{O}",#ファイル番号,"COM:通信パラメータ"
なんて命令を見たことあるぞ。

老：マニュアルを見てなにをいっとんじゃ，この男は。

M：X1turboユーザーズマニュアルですね。なるほど，ちゃんと「RS-232Cの使い方」なんていう章が設けられていますよ。

老：西川よ，ではX-BASICのほうはどんな命令を使うかわかるかな。

善：うーん。zzzz……。

M：寝ちゃいましたよ。

老：起こせ。まったく少し脳味噌を使うとこれだ。X-BASICのファイルオープンの命令ぐらいは知っておろう？

善：ふや？　FOPENだっけ。あ，わかった。FOPEN("COM","R")だ，そうでしょ。

老：ぶぶーっ！　FOPEN("AUX","R")なのじゃ。

善：んなことわかるか！

M：COMはいいとして通信パラメータってなんです？

老：電話で話をするとき，お互い同じ言語を使わないと会話が成り立たんじゃろう？それと同じでパソコン通信を行うときにも2台のパソコンとも同じ条件の下でデータをやりとりしなくてはいかん。その「条件」というのが「通信パラメータ」じゃ。

M：どんなものがあるんですかね。

善：通信速度（ボーレート）やデータ長，パリティビットチェック，ストップビット長，XON/XOFFの指定などがそうじゃ。

老：マニュアルを人の声色で読み上げるな，この愚か者！

M：ボーレートっていうのは昔よく外部記憶装置にカセットテープが使われていた頃によくいわれましたね。

善：あ，知ってる。X1のカセットは2700ボーなんだぜ。

Human68kとデバイス

X68000はHuman68kというDOSを持ち，X-BASICはこのHuman68k上で動作している。DOS上で動作しているということは，そのDOSが提供する各種のサービスをそのまま使えるということを意味するのだ。

CPUとメモリの外についている各種ハードウェアはデバイスドライバというものによってほぼ共通の仕様でアクセスできるように考慮されている。すなわち，ユニファイドI/Oというやつだ。

極端な話，BASIC.CNFでMUSIC.FNC（FM音源制御関数群）を組み込んでいなくても，規定のデータを文字列にして，"OPM"というファイルに書き出してやれば音楽演奏ができる。AD PCMも同様に"PCM"というファイルにデータを書き出せばいい。

プリンタ制御は？　もちろん"PRN"と"LPT"だ。画面出力は"CON"。すると，RS-232Cの制御は当然"AUX"になる。これらの"ファイル"は1文字ずつでも1行ずつでも，好きなように扱える。各種デバイスがこのような方式で制御できるようになっていれば，あらゆるデバイスを制御するのに必要なのはファイル操作関数の使い方だけとなる（そのデバイスに関する知識は必要だが）。

ほかのデバイスと違ってRS-232C関係はBASICの専用関数が用意されていない。だからといって，X-BASICではRS-232C制御ができないと思い込むのは早すぎるのだ。今回の例のように，BASICでもちゃんと通信プログラムを書くことができる。

話は変わるが，先ほど出た「"OPM"に与えるべき」データ。メーカーからは未公開だったものをOh!Xでは独自に「OPMファイル」として扱っていたが，ついにSX-WINDOWで正式な"OPM ファイル"が登場していた。

▶われわれの新潟県にもノッポさんの劇団が来る。うーん，某放送局だけでは生活が苦しいのかな。いいぞー，ノッポさん。ゴンタ君によろしくな！
本間　智（18）X1turbo　新潟県

M：ということはボーレートっていうのはデータの転送速度のことですね。

老：そうじゃ，ふつう75～9600ボーまである。パソコン同士を直結した通信なら9600ボーが高速で便利じゃぞ。

M：データ長というのは？

老：データ長というのはそのデータが何ビットで構成されているかを取り決めるもの，パリティビットというのはまあ，チェックサムみたいなものじゃな。

善：チェックサム？　マシン語ダンプリストなんかによくある……。

老：そのとおりじゃ。パリティには奇数と偶数を指定できるが具体的にどう働いているか説明してつかわそう。たとえばいまパリティビットを奇数と決めて「A」という文字，ASCIIコードにして$41_{H}=01000001_{B}$を送るとパリティビットは1となる。どうしてかわかるかな。

善：zzzzz……。

老：寝るなーっ！　起きろーっ！

善：ふにゃふにゃ1が2つパリティビットと合わせて3つ。

老：そう！　そのとおりじゃ。パリティビットは1データ送るとその後ろにくっつくのじゃが，データ中のビット数とパリティビット自身を足して偶数とするか奇数とするか決めるものなのじゃ。

M：あ，なるほど，だからいまの例では（データ中01000001_{B}の1の数）2＋（パリティビット）1＝3として奇数にしてるわけですね。西川さんさすが。

善：はぁ？（よく意味がわかっていない）

老：逆にいえば（データ中の1の数）＋（パリティビット）は通信パラメータで決めた「偶数/奇数」になっているということじゃ。わかったかな。

M：そうなっていない場合は送受ミスということか。

老：そうじゃ。さて，ストップビットは1データ終了の目印のことじゃが。

善：ちょっと待って。スタートビットはないの？

老：パラメータとしては設定はできないがあることはある。スタートビットは必ず「0」なのじゃ。またストップビットは必ず「1」となる。

善：ほえ？　じゃなんでストップビットなんていうパラメータがあるの？

老：パラメータのストップビットは「0」，「1」を決めるのではなくて何ビット分ストップビットを送るかを決めるものなのじゃ。「1」，「1.5」，「2」の3種類があるな。えー，で，最後のXON/XOFFとは……。

リスト1

```
10 ' R S - 2 3 2 C   U T I L I T Y
20 '
30 '        By Z.NISHIKAWA
40 '
50 ' 通信可能なのはアスキー形式のファイルのみです。
60 ' 通信前に必ず通信パラメータの確認をしましょう。
70 OPTIONSCREEN4:INIT:WIDTH 80,25,0,2:CONSOLE0,25:SCREEN:DEFINTA-Z:KLIST0:KMODE1
80 DIM A$(128),P(128),T(128)
90 DEFFNR$(A$,X)=MID$(A$,X,13)+"."+MID$(A$,X+13,3)
100 DEFFNA(X,Y)=&H2000+X+Y*80
110 DEFFNS$(A)=RIGHT$(STR$(A),LEN(STR$(A))-1)
120 P$="6N81XSLLNZ"     'RS232C ﾊﾟﾗﾒｰﾀ
130 C(1)=2:C(2)=4:C(4)=1:A$=" "
140 RP=32        'ｴﾗｰ発生時のリトライ数
150 FT$=STRING$(13,255)+"."+STRING$(3,255)
160 INPUT "通信パラメータを確認しますか?(Y/N)",A$
170 IF INSTR("Yyﾝ",A$) GOSUB"SW"
180 CLS:KEY 0,""
190 PRINT "[1] 送信モード":PRINT "[2] 受信モード"
200 INPUT "モードを選択して下さい。1"+CHR$(&H1D),MD
210 IF MD=2 THEN "R"
220 CLS
230 COLOR 5:PRINT ">>> 送信モード <<<":COLOR 7
240 SRD$="0:":PRINT"ソースドライブを入力して下さい。(DEFAULT=";SRD$;")"
250 INPUT "SOURCE DRIVE";SRD$
260 IF INSTR(SRD$,":")=0 THEN SRD$=SRD$+":"
270 CLS
280 LOCATE0,0:PRINT" E A S Y   R S 2 3 2 C   U T I L I T Y VERSION 1.00
 BY Z.N
290 '
300 COLOR7
310 Z=0:X$=STRING$(18,&H1D)+CHR$(&H1F)
320 FOR I=16 TO 31
330 DEVI$ SRD$,I,A$,B$
340 FOR J=2 TO LEN(A$)
350 T=ASC(MID$(A$,J-1,1))
360 IF T=0 THEN J=J+31:GOTO390 ELSE T(Z)=T AND 7
370 A$(Z)=FNR$(A$,J):Z=Z+1:J=J+31
380 IF A$(Z-1)=FT$ THEN Z=Z-1:GOTO470
390 NEXT
400 FOR J=2 TO LEN(B$)
410 T=ASC(MID$(B$,J-1,1))
420 IF T=0 THEN J=J+31:GOTO450 ELSE T(Z)=T AND 7
430 A$(Z)=FNR$(B$,J):Z=Z+1:J=J+31
440 IF A$(Z-1)=FT$ THEN Z=Z-1:GOTO470
450 NEXT
460 NEXT
470 X=0:Y=2:'PALET 2,6
480 FOR I=0 TO Z-1
490 COLOR C(T(I))
500 LOCATEX,Y:PRINTUSING"##";I;:PRINT":";:COLOR7:PRINT LEFT$(A$(I),8):Y=Y+1
510 IF Y=23 THEN Y=2:X=X+12
520 NEXT
530 LOCATE22,1:PRINT"コピーしたいファイルを指定して下さい。"
540 LOCATE0,24:COLOR2:PRINT"RED:マシン語 ";:COLOR4:PRINT"GREEN:BASIC ";:COLOR1
:PRINT"BLUE:ASCII";
550 LOCATE0,23:COLOR6:PRINT"[A]=全ﾌｧｲﾙ送信 [S]=送信開始 [SPACE]=決定 [ESC]=
取消 [R]=受信モード
560 COLOR5:CFLASH1:LOCATE38,24:PRINT"X68K MODE =";S68;:CFLASH
570 COLOR7:MX=X
580 X=0:ALL=0:Y=2:K=1:GOTO730
590 '
600 '   SELECTION
610 '
620 A$=INKEY$(0):A=VAL(A$):IF A$="" GOTO620
630 XX=(A=4)-(A=6):YY=(A=8)-(A=2):K=0
640 IF XX+YY THEN COLOR INP(&H2000+X+Y*80) AND 7:IF P(VAL(SCRN$(X,Y,2))) THEN CF
LASH1:LOCATEX,Y:PRINTSCRN$(X,Y,2);:CFLASH:K=1:GOTO710 ELSE LOCATEX,Y:PRINTSCRN$(
X,Y,2);:K=1:GOTO710
650 AA=VAL(SCRN$(X,Y,2))
660 IF A$="A" THEN ALL=1:GOTO780
670 ON INSTR("RS",A$) GOTO "R",780
680 IF INSTR("xXｻ",A$) AND A$<>"" THEN COLOR5:CFLASH1:S68=1 XOR S68:LOCATE38,24:
PRINT"X68K MODE =";S68;:BEEP:CFLASH
690 IF A$=" " OR A$=CHR$(13) THEN IF T(AA)<>4 THEN BEEP ELSE P(AA)=1:A=INP(FNA(X
,Y)) OR 16:POKE@ FNA(X,Y),A,A:YY=1:A$="":GOTO640
700 IF A$=CHR$(27) OR A$=CHR$(8) THEN P(AA)=0:A=INP(FNA(X,Y)) AND 7:POKE@ FNA(X,
Y),A,A:YY=1:A$="":GOTO640
710 IF (X=0 AND XX=-1) OR (X=MX AND XX=1) THEN XX=0 ELSE IF X<68 OR X>12 THEN IF
 SCRN$(X+XX*12,Y,2)="  " THEN XX=0
720 IF (Y=2 AND YY=-1) OR (Y=22 AND YY=1) OR SCRN$(X,Y+YY,2)="  " THEN YY=0
730 X=X+XX*12:Y=Y+YY:IF K THEN COLOR INP(&H2000+X+Y*80) AND 7:CREV1:LOCATEX,Y:PR
INTSCRN$(X,Y,2);:CREV
740 GOTO 620
750 '
760 '   COPY START
770 '
780 CONSOLE23,2:CLS:CONSOLE0,25:LOCATE0,24
790 FOR I=0 TO Z-1
800 IF (ALL=0 AND P(I)=0) OR T(I)<>4 THEN 820 ELSE COLOR5:CSIZE2:PRINT #0,"Copyi
ng "+CHR$(34)+A$(I)+CHR$(34):COLOR 7:CSIZE
810 GOSUB 890
820 NEXT
830 COLOR7:KEY0,"":PRINT:CONSOLE0,25:LOCATE0,23:PRINT"終了しました。";:BEEP
840 OPEN "O",#1,"COM:"+P$
850 PRINT #1,CHR$(&H1A)
```

▶ついに人の道をはずしてしまいました。楠桂にハマッてりぼんコミックに手を出してしまったのです。やっぱり私は，オタッキーなんでしょうか？

佐藤　雄二（17）X1turboZ II　新潟県

善：あ，知ってる。風邪ひいたときやなんかによく出……る。人に噂されているときなんか特に。へえっくそーん（XON）って。なんちゃって，ははは。

老&M：はぁ～。

老：おぬしは編集室でもその調子だそうじゃな。困ったものじゃ。さて，XON/XOFFというのはフロー制御といって……。

善：風呂にはいる順番を決めるものである。なんちゃって。ははは。

M：……西川さん。

老：（まったく無視して）通信速度を速くするとコンピュータがデータを受け取る前に次のデータがきてしまうことがある。これを防止するのがフロー制御じゃ。

善：風呂の湯があふれないようにすると覚えよう。

M：（しつこい人だなぁ）あー。もし通信パラメータが一致していないとどうなるのですか？

老：一度試してみると面白いだろう。ストップビットやパリティなどまでデータと勘違いして受信してしまったり，スタートビットを見失ってしまったりしてデータが滅茶苦茶になるじゃろうな。

通信パラメータの設定

善：通信パラメータの設定の方法はX68000の場合はSPEED.XやSWITCH.Xを実行して行うんだよね。

老：そうじゃ，X1turboの場合はそれぞれのパラメータの意味に対応した文字列によって設定してやるのじゃ。詳しくはユーザーマニュアルを見たほうがいいじゃろう。

M：あれ？　これだけで次の話にいっちゃうの？

データ送受の仕方

老：さて，2台のパソコンのパラメータを同じに揃えたら今度は実際の入出力じゃ。

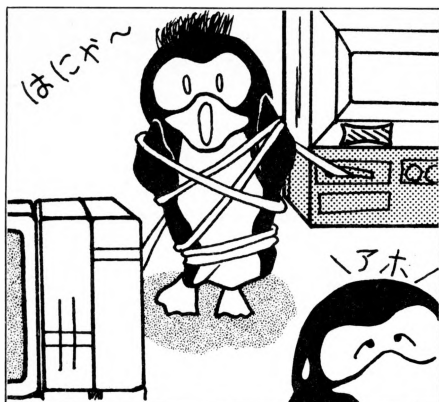
M：ディスクなんかのときにはX1turboでは「PRINT　#（出力）」，「INPUT　#（入力）」を使い，X68000の場合は「FWRITES（出力）」，「FREADS（入力）」を使いましたね。

老：実はRS-232Cへ出力するときもほとんどディスクやテープに出力するのと同じ方法でいいのじゃ。

善：「，」なんかも入力してくれる「LINPUT　#」やなんかも使えるわけ？

老：そうじゃ。では，通信が終わったらなにをしたらいいかはもういわなくてもわか

```
860 CLOSE
870 END
880 '
890 '   TRANSFER ROUTINE
900 '
910 NL=0:F68$=""
920 OPEN "O",#2,"COM:"+P$
930 OPEN "I",#1,SRD$+A$(I)
940 GOSUB 1070
950 PRINT #2,F68$
960 LINPUT #1,A$:IF LEN(A$)=0 THEN NL=NL+1:IF NL<RP GOTO 960 ELSE GOTO 1040
970 ON S68 GOSUB 1220
980 IF INSTR(A$,"DATA") AND S68 THEN GOSUB 1190
990 IF INSTR(A$,"' ") AND S68 THEN GOSUB1160:GOTO1010
1000 IF INSTR(A$,"'") AND S68 THEN GOSUB 1260
1010 KMODE 0:PRINT  #2,A$
1020 KMODE 1:LOCATE0,24:PRINT#0,A$
1030 GOTO960
1040 PRINT #2,"EOF"
1050 CLOSE:PAUSE 10
1060 RETURN
1070 'REMAKE FILE NAME
1080 B$=LEFT$(A$(I),8)+"."+RIGHT$(A$(I),3)
1090 FOR J=1 TO LEN(B$)
1100 A$=MID$(B$,J,1)
1110 IF ASC(A$)>&H20 THEN F68$=F68$+A$
1120 NEXT
1130 IF MID$(F68$,LEN(F68$),1)="." THEN F68$=LEFT$(F68$,LEN(F68$)-1)
1140 RETURN
1150 'REM ﾌﾞﾝ ｦ ﾅｵｽ 1
1160 E=INSTR(A$,"' ")
1170 MID$(A$,E,2)="/*":RETURN
1180 'DATAﾌﾞﾝｦ ｹｽ
1190 E=INSTR(A$,"DATA")
1200 MID$(A$,E,4)="    ":a$=a$+",":RETURN
1210 'ｷｮｳﾊﾞﾝｺﾞｳｦ ｿﾛｴﾙ
1220 E=INSTR(A$," ")
1230 A1$=MID$(A$,E,255)
1240 A$=RIGHT$("    "+LEFT$(A$,E-1),5)+A1$:RETURN
1250 'REM ﾌﾞﾝ ｦ ﾅｵｽ 2
1260 E=INSTR(A$,"'")
1270 A$=LEFT$(A$,E-1)+"/*"+MID$(A$,E+1,255):RETURN
1280 '
1290 '    EASY RS232C CONVERTER (FROM 68 TO X1)
1300 '
1310 LABEL"R"
1320 KEY 0,"":CONSOLE 0,25:CLS
1330 COLOR 6:PRINT">>> 受信モード <<<":COLOR7
1340 LN=10
1350 INIT"MEM:"
1360 PRINT ">>> 受したデータをﾌｧｲﾙとしてG-RAMに作成します。";
1370 PRINT "そのﾌｧｲﾙ名は何にしますか? <<<"
1380 FL$="TEMP"
1390 INPUT "FILE NAME:"+FL$+STRING$(LEN(FL$),&H1D),FL$
1400 '
1410 OPEN "I",#1,"COM:"+P$
1420 OPEN "O",#2,"MEM:"+FL$
1430 TP=2:PRINT
1440 PRINT ">>> 受信するファイルのタイプ <<<
1450 PRINT "0. ドキュメント・ファイル (ED.X等の文書)
1460 PRINT "1. BASICファイル
1470 PRINT "2. ドキュメント・ファイルに行番号等を付けてBASICファイルを
作る。
1480 INPUT TP
1490 IF TP=0 GOTO1540
1500 PRINT:PRINT ">>> 受信データはREM文にするか、DATA文にするかを決定
して下さい。<<<"
1510 INPUT "[REM] OR [DATA] (R/D):",A$
1520 IF A$="D" OR A$="d" THEN G$=" DATA " ELSE G$=" ' "
1530 '
1540  LINPUT #1,A$:IF LEN(A$)=0 THEN A$=A$+CHR$(&HD,&H1A,0,0)
1550  ON TP GOSUB 1640,1760
1560  PRINT  #2,A$
1570  LOCATE0,24:PRINT#0,A$
1580  L=LOC(1):'LOCATE0,0:COLOR5:PRINTL:COLOR7
1590  IF L=1 GOTO1610
1600  GOTO1540
1610 CLOSE
1620 END
1630 '>>> BASIC FILE <<<
1640 IF MID$(A$,1,1)=" " THEN A$=MID$(A$,2,255):GOTO 1640
1650 L$=""
1660 FOR I=1 TO LEN(A$)
1670 B$=MID$(A$,I,1)
1680 IF B$<="9" AND B$>="0" THEN L$=L$+B$ ELSE 1700
1690 NEXT
1700 IF VAL(A$)=0 THEN PRINT"FILE TYPE IS DIFFERENT.":END
1710 IN=INSTR(A$,L$)
1720 A$=MID$(A$,IN+LEN(L$),255)
1730 A$=L$+G$+A$
1740 RETURN
1750 '>>> MAKE BASIC FILE FROM DOCUMENT <<<
1760 A$=FNS$(LN)+G$+A$
1770 LN=LN+10:RETURN
1780 '
1790 '  PARAMETER
1800 '
1810 LABEL"SW"
```

▶皆さん，桜は咲いたでしょうか？　私はなんとか咲きました(私立は全滅でしたが)。
石川　貴康（18）X68000　富山県

るじゃろう。

M：X68000はFCLOSEALL( )やFCLOSE。

善：X1はCLOSEだな。

老：そのと一り。では，RS-232Cで通信プログラムを組む手順を整理してみなさい，マスター。

M：はい，

1) 通信パラメータを送信側，受信側で一致させる。

2) AUX，COMなどでファイルをオープンする。

3) FREADS，FWRITES，LINPUT，PRINTなどで送受信を行う。

4) FCLOSE，CLOSEでファイルをクローズする。

サンプルプログラム

数日後……

善：このあいだ教わった手順で簡単なプログラムを作ってみたんですが（リスト1：X1turbo用，リスト2：X68000用）。

老：ほう。どれどれ。なるほど，X1turboが送信側の場合はメニューでファイルを選んで送信するのか。X68000が受信側の場合は簡単な受信プログラム（リスト2）が必要なのじゃな。

M：リスト1には受信プログラムや，通信パラメータの簡単なエディタもついているようですね。あれ，X1側には送信，受信のプログラムがあるようですがX68000側は受信プログラムだけなんですか？

善：X68000はOSがしっかりしているのでOSのCOPYコマンドで送信できちゃうんだ。具体的には，

A＞COPY　ファイルネーム　AUX

とやればよい。

老：うーむ。受信するときもリスト2なんぞ使わず，

A＞COPY　AUX　ファイルネーム

で行けると思うが。

善：いやぁ，X1側の漢字を含んだファイル

```
1820 R$(1)=" 偶 数 ":R$(2)=" 奇 数 ":R$(3)=" な し ":RR$(1)="E":RR$(2)="O":RR$(3)="N"
1830 S$(1)="1ﾋﾞｯﾄ  ":S$(2)="1.5ﾋﾞｯﾄ":S$(3)="2ﾋﾞｯﾄ  "
1840 X$(1)="XON":X$(2)="RTS":X$(3)=" な し ":XX$(1)="X":XX$(2)="R":XX$(3)="N"
1850 X=10:Y=2:K=1
1860 B=VAL(MID$(P$,1,1))
1870 T=VAL(MID$(P$,3,1))
1880 R=INSTR("EON",MID$(P$,2,1))
1890 S=VAL(MID$(P$,4,1))
1900 F=INSTR("XRN",MID$(P$,5,1))
1910 CLS:COLOR 6:PRINT">>> 通 信 パ ラ メ ー タ の 設 定 <<<":COLOR 7
1920 PRINT
1930 PRINT "ﾎﾞｰﾚｰﾄ":PRINT "ﾃﾞｰﾀ 長 ":PRINT "ﾊﾟﾘﾃｨ":PRINT "ｽﾄｯﾌﾟ ":PRINT " 通 信 制
御 "
1940 COLOR 5:LOCATE 0,8:PRINT"[ESC]=ﾃﾞﾌｫﾙﾄ 値 に 戻 す  [RET]= 終 了 [TENKEYS]=ｶｰｿﾙ
移 動、ﾊﾟﾗﾒｰﾀ 変 更 ":COLOR 7
1950 GOSUB 1960:GOTO 2130
1960 LOCATE  9,2:PRINT INT(150*2^B):MID$(P$,1,1)=FNS$(B)
1970 LOCATE 10,3:PRINT FNS$(T);"ﾋﾞｯﾄ":MID$(P$,3,1)=FNS$(T)
1980 LOCATE 10,4:PRINT R$(R):MID$(P$,2,1)=RR$(R)
1990 LOCATE 10,5:PRINT S$(S):MID$(P$,4,1)=FNS$(S)
2000 LOCATE 10,6:PRINT X$(F):MID$(P$,5,1)=XX$(F)
2010 RETURN
2020 A$=INKEY$(0):A=VAL(A$):XX=(A=4)-(A=6):YY=(A=8)-(A=2):K=0
2030 IF YY THEN COLOR INP(&H2000+X+Y*80) AND 7:LOCATEX,Y:PRINT SCRN$(X,Y,7);:K=1
:GOTO2110
2040 IF A$=CHR$(27) THEN P$="6N81XSLLNZ":GOTO1820
2050 IF A$=CHR$(13) OR A$=" " THEN RETURN
2060 IF XX AND Y=2 AND (B+XX>=0) AND (B+XX<=6) THEN B=B+XX:GOSUB 1960:K=1:GOTO21
30
2070 IF XX AND Y=3 AND (T+XX>=5) AND (T+XX<=8) THEN T=T+XX:GOSUB 1960:K=1:GOTO21
30
2080 IF XX AND Y=4 AND (R+XX>=1) AND (R+XX<=3) THEN R=R+XX:GOSUB 1960:K=1:GOTO21
30
2090 IF XX AND Y=5 AND (S+XX>=1) AND (S+XX<=3) THEN S=S+XX:GOSUB 1960:K=1:GOTO21
30
2100 IF XX AND Y=6 AND (F+XX>=1) AND (F+XX<=3) THEN F=F+XX:GOSUB 1960:K=1:GOTO21
30
2110 IF (Y=2 AND YY=-1) OR (Y=22 AND YY=1) OR SCRN$(X,Y+YY,2)="  " THEN YY=0
2120 Y=Y+YY
2130 IF K THEN COLOR INP(&H2000+X+Y*80) AND 7:CREV1:LOCATEX,Y:PRINTSCRN$(X,Y,7);
:CREV
2140 GOTO 2020
```

リスト2

```
 10 width 96
 20 char A,B,b,c
 30 int i
 40 dim char t(0)
 50 str  S[256],T
 60 while 1
 65      c=0
 70      repeat        /* ごみデータの削除
 80        A=fopen("AUX","R")
 90        fread(t,1,A)
 91        if t(0)=&H1A then c=c+1:if c>1 then end
100      until t(0)>&H20
110   T=chr$(t(0))
120   freads(S,A)       /* FILE NAME 受信
130   S=T+S
140   color 6
150   print "Receiving ";S
160   color 3
170   B=fopen(S,"c")
171   if B<0 then print "ファイルネームに異常有り!":wait():beep:end
180    while 1
190      freads(S,A)
200      if S="EOF" then fwrites(chr$(&H1A),B):fcloseall():break
210      knj()          /* 漢字を含まないファイルを受信するならばここは消してもOK
220      print S
230      S=S+chr$(13)+chr$(10)
240      fwrites(S,B)
250    endwhile
260 endwhile
270 end
280 func knj()          /* 漢字データの修正
290 for i=1 to len(S)
300   b=asc(mid$(S,i,1))
310   if b=&HE or b=&HF then S=left$(S,i-1)+right$(S,len(S)-i)
320 next
330 endfunc
340 func wait()
350 A=val(right$(time$,2))
360 repeat
370   B=val(right$(time$,2))
380   if B<A then B=B+60
390 until B-A>=4
400 endfunc
```

▶たのむからサークル紹介載せてくれ。サークル名は，PROFESSIONAL-GAMERだ。
菅田　朋樹（16）X68000PRO，X1F　富山県

をX68000で受信すると文字化け（要するに受信ミス）が起こるんだな。原因はX1が漢字IN/OUTのコードまで送ってしまうからなんですね。リスト2の後ろにその対応サブルーチンがついているでしょ。

老：なるほど。漢字を含まないファイルならわしがいった方法でもいいのかな。

善：ええ，だけどそうすると，ファイルネームや「EOF」という文字列も受信してファイルに書いてしまうので一度エディタ(ED.Xなど）でそれらの文字列を削除しなくてはいけないのでなるべくこちらを使ったほうがいいと思うけど。

M：リスト1の送信側の「X68K MODE」っていうのはなんです？

善：あぁ，それはね。僕はよくBASICプログラムや音楽プログラムを転送することが多いのでX1側でASCIIセーブしたBASICファイルをそのままX-BASICで読み込めるように変換するコマンドだよ。「X68K MODE」＝1でX1BASICの「 ' 」を「/*」に，「DATA文」をスペースに変化したり，行番号をX-BASICのフォーマットに直してくれるよ。このモードの切り替えは[X]キーを使う。

M：リスト2はCコンパイラでコンパイルしたらもっと速くなりそうですね。（当たり前だな）

善：そうそう，X1ではファイルネームにどんな文字を使っても大丈夫だったのですがX68000ではある程度制限されるのでそのへんはご了承くださいナ。変なファイルネームではエラーが出たり，ファイルの作成に失敗することがありますよ。

M：X形式などやサンプリングデータなどのオブジェクトデータは通信できないのですか？

善：できません。いやできないことはないのだけれどあまり必要ないかなぁ，と思ったのでつけなかった。

老：それは日本語では「手抜き」，英語では「ハンドピック」というんだぞ。（ほんとかおい！）

善：えーん。安ドル高。

M：この人は……。

老：まぁ，よい。ひとつの解決策としては16進データを文字型データに変換してから転送し，それから元に戻すという手が挙げられる。たとえばリスト3や4のようなもので文字型データにするとよいじゃろう。元に戻すほうはとても簡単だから各自の自由研究としよう。

一同：じゃあ，またZ80'sBARで会おうね。ばぁーい。

リスト3

```
10 DEFINTA-Z:WIDTH 80:INIT
20 DEFFNR$(A)=RIGHT$(STR$(A),LEN(STR$(A))-1)
30 CM$=","
40 INPUT"START:&H",S$:INPUT"END  :&H",E$
50 INPUT"STEP :",G:IF G>15 THEN G=16 ELSE G=8
60 INPUT"LINE :",LN:IF LN<10 THEN LN=10
70 INPUT"D)ATA or R)EM:",A$:IF A$="R" OR A$="r" THEN G$="' " ELSE G$="DATA "
80 INPUT"SAVE TO DEVICE? [Y/N]",YN$:IF YN$="Y" GOSUB 160
90 S=VAL("&H"+S$):E=VAL("&H"+E$)
100 FOR I=S TO E STEP G:SM=0:PRINTLN;G$;:SV$="":IF YN$="Y" THEN SV$=MID$(STR$(LN
),2,LEN(STR$(LN))-1)+G$
110 FOR J=0 TO G-1:P=PEEK(I+J):PRINT FNR$(P);CM$;:IF YN$="Y" THENSV$=SV$+FNR$(P)
+CM$
120 NEXT:LN=LN+10:PRINTCHR$(&H1D);" ";:IF YN$="Y" THEN SV$=LEFT$(SV$,LEN(SV$)-1)
130 PRINT:IF YN$="Y" THENPRINT#1,SV$
140 NEXT:IF YN$="Y" THEN CLOSE
150 END
160 INPUT"DEVICE=",DV$:INPUT"FILENAME:",FL$:FL$=LEFT$(FL$,13)
170 OPEN "O",#1,DV$+":"+FL$:RETURN
```

リスト4

```
  10 /* FILE BIN DATA CONVERT TO HEX STR DATA
  20 width 96
  30 dim char D(65535)
  40 char FN1,FN2
  50 str S[256],f1[256],f2[256],a
  60 int dmy,how,I,J
  70 /*
  80 print "16進データを文字列に変換します。"
  90 input "転送元FILE NAME:",f1
 100 if f1="" then bye()
 110 input "転送先FILE NAME:",f2
 120 if f2="" then bye()
 130 print "ファイルの何バイト目から何バイト分変換しますか？(RETURNキ-連打でファイル
まるごと)
 140 input "何バイト目から(0〜):&H",a
 150 dmy=val("&h"+a)
 160 if dmy>65535 then bye()
 170 input "何バイト分変換(1〜):&H",a
 180 how=val("&h"+a)
 190 if how>65535 then bye()
 200 FN1=fopen(f1,"r")
 210 FN2=fopen(f2,"c")
 220 if dmy then fread(D,dmy,FN1)
 230 if dmy=0 and how=0 then get_how()
 240 if how then fread(D,how,FN1) else bye()
 250 I=0
 260 repeat
 270 S=""
 280 for J=I to I+15
 290 S=S+right$("0"+hex$(D(J)),2)+" "
 300 next
 310 print S,hex$(I)
 320 S=S+chr$(13)+chr$(10)
 330 fwrites(S,FN2)
 340 I=I+16
 350 until I>=how
 360 S=chr$(&H1A)
 370 fwrites(S,FN2)
 380 fcloseall()
 390 end
 400 func bye()
 410 end
 420 endfunc
 430 func get_how()
 440 how=fseek(FN1,0,2)
 450 fseek(FN1,0,0)
 460 endfunc
```

RS-232Cと通信

現在では通信といえばRS-232Cというふうに，パソコンの一般的なデータ転送の手段としてRS-232Cが採用されている。一般のRS-232C回線が扱っているのはテキストデータ（文字だけのデータ）だ。制御コードと同じデータを持ち得るオブジェクトコードを直接転送することはできない。

オブジェクト用の通信方式としてX-MODEMやY-MODEMなどの方式もあるが，パソコン通信の場合，オブジェクトはISHと呼ばれる方式でテキスト形式に変換されることが多い。ISHは効率のよい変換を行い，通信回線の不調でビット落ちなどが起こっても少々なら修復してしまう。パソコン通信を始めようという人はまずなんとかしてISHを手に入れること。あとはどうにでもなる。

元々汎用規格のRS-232Cの適用範囲はパソコンに限らない。あらゆるものがRS-232Cを通したデータ交換の対象となる。最近のポータブルワープロは通信機能を備えているものも多くなってきたから，極端なところでは，ポケコンのプログラムをポータブルワープロのフロッピーディスクにバックアップする，といったことも可能かもしれない。

▶いま私は，非常に悲しい。command.xより大きくなったピコピコゲームの入ったハードディスクをぶっ飛ばしてしまったからである。「初期化しますか？」「n」としたのに，ハードディスクはすべて未確保の領域に。現在PM11:30……。
佐渡　詩郎（14）X68000ACE-HD　石川県

［特集］
BASICプログラミング

BASIC

アルゴリズムを考える

拡大縮小処理の基本

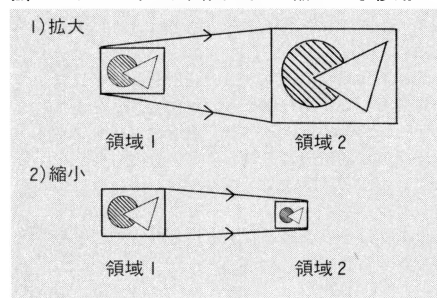
ラインルーチンのアルゴリズムによるグラフィックの拡大縮小処理を考えます。BASICが遅いというのは常識ですが，その代わりBASICならどんな処理も簡単にこなせます。遅いなら遅いなりにアルゴリズムの改良による高速化を行ってみましょう。

Tan Akihiko
丹 明彦

X68000のゲームのなかでも突出したジャンルは，あのスペースハリアーに始まる3Dものであろう。大小さまざまのキャラクタが画面内をえらいスピードで乱れ飛んでいるのを見て，驚かなかった者はいない。いったいどのようにして実現しているのか，不思議に思った方もあるだろう。

周知の事実として，かどうかは知らないが，はるか彼方の小さなキャラクタから目の前いっぱいに広がる大きなキャラクタまで，画面にはいろいろな大きさのキャラクタが出現する可能性がある。だが，それぞれの大きさのパターンをひとつひとつキャラクタエディタで描いているわけでは決してない。たとえば極端な話，ナイトアームズで32768段階のパターンをすべて別々に用意していたのでは，プログラマもデザイナーも，それにフロッピーディスクやメモリも，たまったものではなかろう。この膨大な量のデータに，もっとうまい方法で対処しているはずなのである。

で，拡大縮小PUTルーチンである。自機にしろ敵機にしろ，用意しておく基本パターンはひとつだけ。あとはそのパターンに拡大（縮小）率をくっつけて呼び出せば，どんな大きさのキャラクタでも画面に出せる。それが拡大縮小PUTルーチンだ。

僕自身3D体感ゲームの開発にかかわったわけでもなんでもないので，これから紹介するアルゴリズムが現実にゲームで使われているアルゴリズムと同じものかどうかは保証しないが，原理はとても簡単で，グラフィック命令のある言語なら必ず作れる処理である（いざ高速化しようと思ったらとたんに難しくなる処理でもあるが）。

そのつもりで作り始めたのだが，いつのまにかキャラクタの拡大縮小という当初の目的からは次第にそれてゆき，単なるグラフィック画面の拡大縮小つき移動ルーチンになってしまった（ありがちな展開だこと）。これはひとえに，キャラクタのパターンを用意するのが面倒になったという安易な理由のせいである。

BASICなので速度はたいして期待していなかったが，これがまた予想外に遅いできばえである。そう，このあきれるほどのスピードも拡大縮小PUTルーチンにできなかった理由のひとつであった。

それでも，せめてもの抵抗をしていくなかで，インタプリタ向けの高速化とコンパイラ向けのそれとでは作法が微妙に違うことがわかった。これは収穫であった。

図1　グラフィック画面の拡大縮小つき移動

まずは原理から

拡大縮小の原理は簡単である。グラフィック画面上の矩形領域1（矩形は長方形という意味）を矩形領域2に拡大（または縮小）して転送することを考える（図1）。2つの領域はともにx，y軸と平行に置いてある。これらは相似図形とは限らない（縦横比が違っていても構わない）。図2が今回作る拡大縮小ルーチンの仕様である。

ここで約束ごとをひとつしておくことにしよう。以後，領域1を表す変数には“f”という添え字がつく。これは領域1「から」転送するという意味をこめて“from”から取ったものである。同様に，領域2は“to”の“t”を添え字としてつける。

ではさっそく図3をご覧いただこう。領域1をx方向にはt_x：$(1-t_x)$に内分し，y方向にはt_y：$(1-t_y)$に内分する点(x_f, y_f)と，領域2を同じ比で内分する点(x_t, y_t)とは対応しているので，同じ色になるはずである。したがって，領域内のすべての点について，

pset(x_t, y_t, point(x_f, y_f))

としてやれば転送が可能である。これが拡大縮小アルゴリズムのすべてである。これだけを予備知識として持っておけば，BASICでプログラムを書くことはできる。

が，机の上で考えることと現実のプログラムの間には，往々にしてギャップがあるものである。そのギャップを埋められるかどうかが，プログラムを書ける人間かそうでないかの分かれ目になるのではないか，僕はそう思っている。コマンドや関数を覚え込む必要などない。そんなものは使っているうちに覚えていくものである（たとえばCライブラリのマニュアルは数百ページもあるが，実際僕が覚えているのはその中のほんの数ページである。あとは必要になったらマニュアルをひっくり返すだけで用が足りる。よく使うものは自然に覚えていくので問題はない）。今回使う関数の中で主役級といえるのは，先ほどもちょっと出てきた，

pset(x, y, c)…(x, y)に色cを打つ
point(x, y)…(x, y)の色を調べる

の2つだけである。大切なのは，使う言語が変わっていってもきちんと対処できるような作法を身につけることである（うーむ今回は説教調だ）。

まず誰でも考えるのが，「領域1内の各点を領域2に移す」という手法であろう。それがリスト1である。

領域1の座標（x_f, y_f）で2重のforループを作り，（x_f1, y_f1）と（x_f2, y_f2）を使ってt_xとt_yを逆算する（簡単な1次方程

▶4月号の「限りなき戦い」を読んで，小学生のころこのゲームをやっていたことを思いだした。懐かしいなぁ。
江野　尊治（18）X1C　石川県

式で解ける)。そのt_xとt_yから領域2の座標(x_t, y_t)を求めることもまたやさしい。で,(x_f, y_f)の色を拾い上げ,(x_t, y_t)に打つ。なんだか言葉でやるとゴチャゴチャだな。BASICプログラムのほうがよっぽどスッキリしている。

これでうまくいくと思ったらどっこいそうは問屋が卸さない。リスト1を実行してみると,拡大が,明らかに妙な実行結果になる。表示が飛び飛びになって,すきま風が吹いてしまっている。あたかもMZ-80Kシリーズのセミグラフィックのような……。それにしてもなぜだ? 水も漏らさぬ論理だと思ったのに。

結論から先にいうと,「領域1」の点についてループを作ったのがそもそもの間違いだったのである。本当は,「領域2の各点に対応する領域1の点を拾ってくる」ようにしなくてはならなかったのである(この違いわかるかな)。ちょっと考えればすぐわかる。

なぜって,目的は「領域2」に表示することではなかったか。だから,領域2の点にすべてプロットするために,領域2の点についてループを作るべきである。領域1の点をくまなく,一度ずつアクセスする意味はまったくない,必要なだけアクセスすればよい。「くまなく一度ずつ」は領域2に対してこそ行うべきだ。以上の問題点を修正したのがリスト2である。今度はうまくいったことと思う。めでたしめでたし。

ちなみに,拡大縮小は変換操作の一種だ(このプログラムの場合は平行移動が入るので1次変換とは呼べないけれども,座標変換であることに変わりはない)が,実はプログラムのうえでは逆変換(つまり領域2→領域1の座標変換)を行わなければいけないことは,CGにおけるマッピングの基本といってもいい。

図2 今回作る転送ルーチンの仕様

●転送元と転送先は,特に相似である必要はない。つまり,x方向とy方向の拡大率は自由に設定できる。

●領域の指定は左上から右下へ行う。つまり,

$x_f1 < x_f2$

$y_f1 < y_f2$

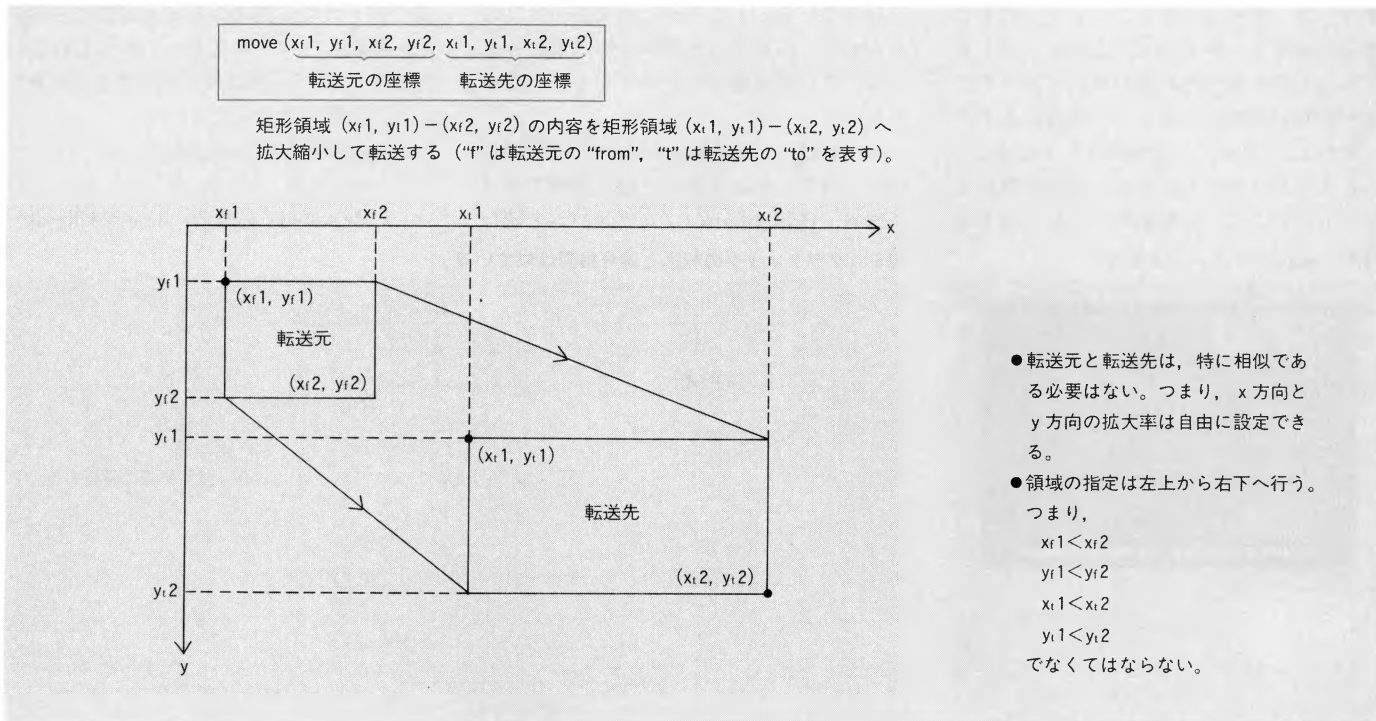
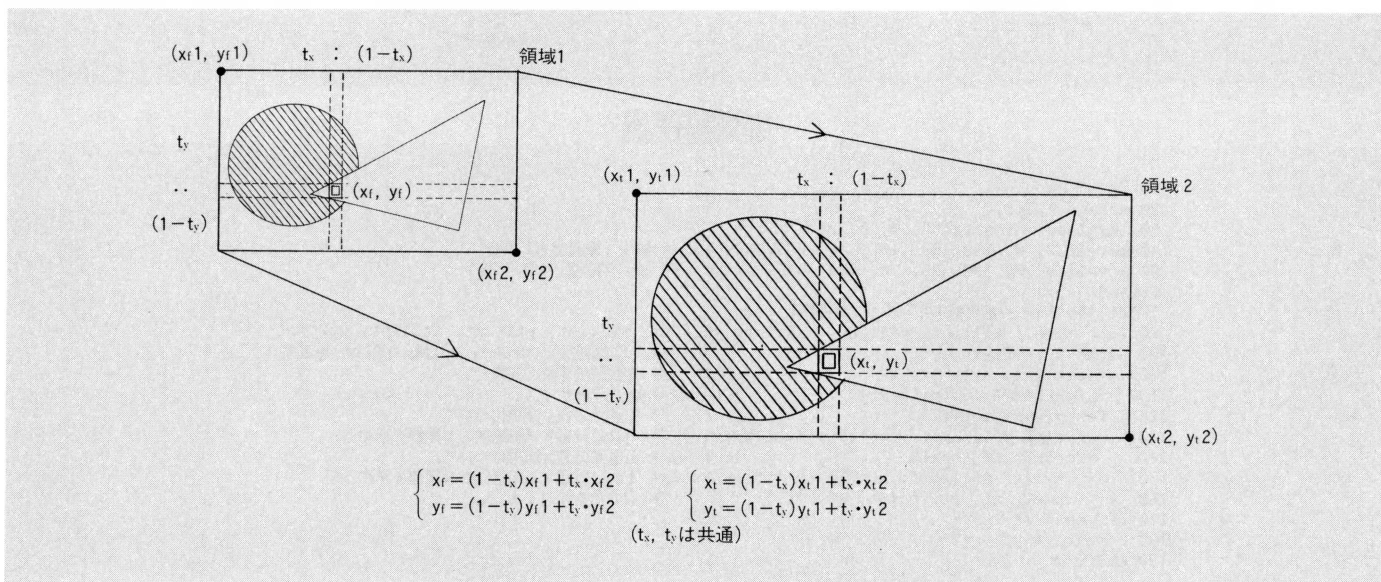
$x_t1 < x_t2$

$y_t1 < y_t2$

でなくてはならない。

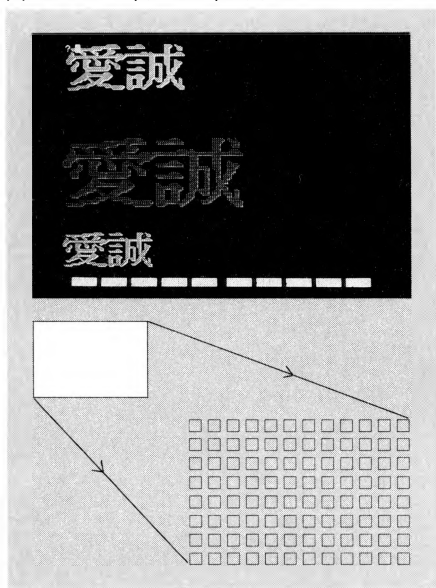
図3 考え方

$$\begin{cases} x_f = (1-t_x)x_f1 + t_x \cdot x_f2 \\ y_f = (1-t_y)y_f1 + t_y \cdot y_f2 \end{cases} \quad \begin{cases} x_t = (1-t_x)x_t1 + t_x \cdot x_t2 \\ y_t = (1-t_y)y_t1 + t_y \cdot y_t2 \end{cases}$$

(t_x, t_yは共通)

高速化する

ここでお開きにしてもいいくらいだが，実行してみると，けっこう我慢ならない遅さである。というわけで，ここからもっと速くしたい。そんなとき，僕らはどこに目をつけたらよいのだろうか。まず一般論からいってみよう。

まず実数が使ってある部分では，できる限り整数を使って「うまく」書き直すようにすること。これは，整数演算のほうがコンピュータにとっては取り扱いが楽であることによる。それから割り算と掛け算も，やはり足し算や引き算で「うまく」表現できないか考えてみること。これも，いうまでもなく割り算や掛け算がコンピュータにとって重い処理であるという理由による。

ただし，「うまく」と強調したのには，それなりにわけがある。単純に実数を整数に変えるだけでは，計算誤差が大きくなる可能性があることがひとつ。また，アルゴリズムの種類によっては素直に実数や掛け算を使ったほうがはるかに速くなるという現象もある。このへんはもう，経験だけが頼りといえるかもしれない。

それぞれのアルゴリズムにもっとも適した高速化はプログラマの裁量で決まるもので，ときにはアルゴリズムの心臓部にまでメスを入れる決断もプログラマには求められる。

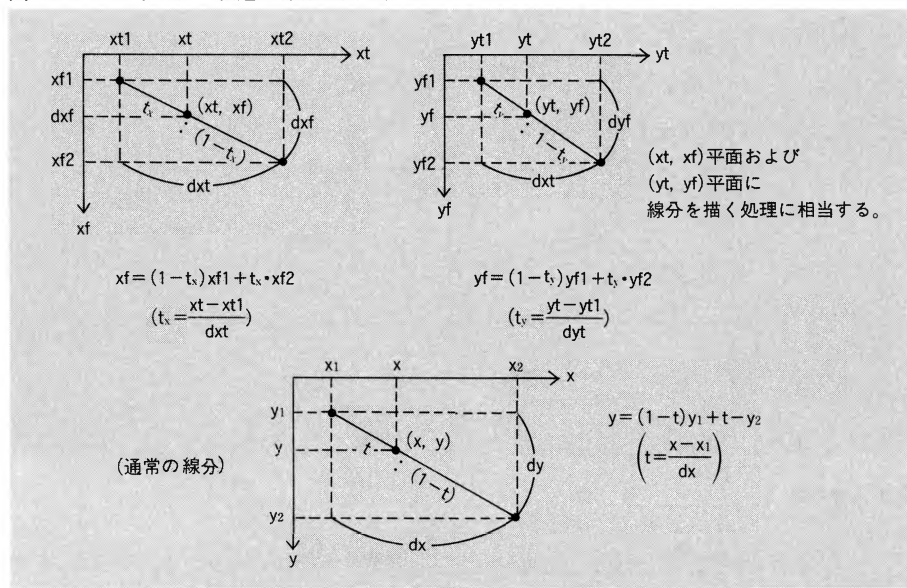
そこで本題の拡大縮小ルーチンの高速化である。またも結論先行になるが，これは線分の描き方によく似ている。図5をご覧いただこう。線分はxからyを求める形式になっている。ところが，拡大縮小は，x_tからx_fを，y_tからy_fを求める形式になっていて，式の形は線分のそれとまったく変わらない。

とすれば，手前味噌な話ながら，僕が1989年7月号で解説を試みた線分描画アルゴリズム，Bresenham（ブレゼンハムと読むらしい）のアルゴリズムを転用できそうである。Bresenhamアルゴリズムは，すべて整数変数で，また足し算と引き算と少しの掛け算で，正確に線分を描画することができる。ということは，いままで使っていた実数変数（floatで宣言していた変数）を排除できることになる（脱線になるが，リスト2のままでも，うまくやれば整数だけで処理はできるが，計算誤差の問題は免れないだろう）。どうやら高速化が望めそうだ。

ここでBresenhamアルゴリズムをちょっと復習しておこう。原理は昔説明したのでもうしない。プログラムを書き換えるために必要な部分だけを抜き出す。X-BASIC形式で書くので，興味のある方は標準装備のline関数と比べてみてもいいかもしれない。ただし，このline0関数には不備な点があるので，実用にはならない。

```
func line0(x1,y1,x2,y2,c)
  int x, y, dx, dy, e
  dx=x2-x1: dy=y2-y1: e=2*dy-dx
```

図4　move1(リスト1)の失敗

図5　グラフィックの転送と線分描画は似ている

リスト1

```
 10 /*   リスト1
 20 screen 1,3,1,1
 30 img_load( "test.gl3", 0, 0, 0 )
 40 move1( 0, 0, 199, 99,  0, 200, 299, 349 )  /* 拡大(縦横とも1.5倍)
 50 move1( 0, 0, 199, 99,  0, 400, 149, 474 )  /* 縮小(縦横とも0.75倍)
 60 end
 70 /* グラフィック拡大縮小ルーチン(失敗版)
 80 func move1( xf1;int, yf1;int, xf2;int, yf2;int, xt1;int, yt1;int, xt2;int, yt2;int )
 90   int xf, yf, xt, yt                        /* "f"は転送元(from),"t"は転送先(to)を表す
100   float dxf, dyf, tx, ty                    /* 実数型の変数の宣言
110   dxf=xf2-xf1: dyf=yf2-yf1                  /* 転送元のサイズ
120   for yf=yf1 to yf2                         /* 転送元の各y座標について
130     ty=(yf-yf1)/dyf: yt=(1#-ty)*yt1+ty*yt2  /* それに対応する転送先のy座標を求める
140     for xf=xf1 to xf2                       /* 転送元の各x座標について
150       tx=(xf-xf1)/dxf: xt=(1#-tx)*xt1+tx*xt2/* それに対応する転送先のx座標を求める
160       pset( xt, yt, point( xf, yf ) )       /* 点の色を転送する
170     next /* xt
180   next /* yt
190 endfunc
```

```
    y=y1
    for x=x1 to x2
      pset(x, y, c)
      while ( e>=0 )
        y=y+1
        e=e-2*dx
      endwhile
      e=e+2*dy
    next
  endfunc
```

これを応用して，リスト2を書き直したのがリスト3。float宣言も割り算もなくなっていることにご注意。

はたして高速化の効果はいかに。なんてこったい，期待に反してかなり遅くなってしまったではないか。なぜだろう。原因はいろいろ考えられるが，いちばんきいているのは行数が長くなってしまったことであろう。そう，インタプリタとコンパイラの差はこういうところで顕著に現れる。

インタプリタは，プログラムを1行ずつ解釈して実行する。そのため，実数を使っていてもシンプルにまとめたプログラムは速く，実数を使わなくてすんでいても命令数が多いプログラムの場合は実行よりも解釈のほうに時間をよけい取られてしまい，結果的に遅くなることになる。ただ整数化しても速くなるとは限らない好例である。こんなタコな話はない。

しかし高速化すると宣言した以上，引っ込みはつかない。うまくいかなくてもあきらめてはいけない。転んでもタダでは起きないくらいの根性を持とう。この失敗作をなんとかして速くしてやろうではないか。そう思って改めてリスト3を見る。

座標計算に明らかに無駄があるのがおわかりいただけるだろうか。y_tはy_t1からy_t2まで1ずつ増えていく。これはいい。対応するy_fは，1回ずつしか計算されないからだ。ところが，x_fの計算は毎回同じことをしている。合計(dy_t+1)回も同じ結果の出る計算をしている勘定になる。これはとんでもない無駄である。これをなんとかして1回に抑えれば，きっとリスト2よりも速いルーチンができあがるであろう。しかしどうすればいいのだろう。

仮に，僕が大量の計算を手でやるという非人間的なことを命じられたとしよう。こんなときは，少しでもラクしようというのが人情だ。もしちょっとでも同じことの繰り返しになる計算が出現したら，その結果をノートに書きとめておくのが賢い。次からはそれを見て書き移すだけ。だいぶ得した気分になれるし，作業も速くなるし，計算ミスも減る。コンピュータだって同じである(もっとも計算ミスはしないだろうが)。繰り返しになる計算結果はどこかに記憶し

リスト2

```
 10 /*   リスト2
 20 screen 1,3,1,1
 30 img_load( "test.gl3", 0, 0, 0 )
 40 move2( 0, 0, 199, 99,  0, 200, 299, 349 )  /* 拡大(縦横とも1.5倍)
 50 move2( 0, 0, 199, 99,  0, 400, 149, 474 )  /* 縮小(縦横とも0.75倍)
 60 end
 70 /* グラフィック拡大縮小ルーチン(改訂版)
 80 func move2( xf1;int, yf1;int, xf2;int, yf2;int, xt1;int, yt1;int, xt2;int, yt2;int )
 90   int xf, yf, xt, yt
100   float dxt, dyt, tx, ty
110   dxt=xt2-xt1: dyt=yt2-yt1                     /* 転送「先」のサイズ
120   for yt=yt1 to yt2                            /* 転送「先」の各y座標について
130     ty=(yt-yt1)/dyt: yf=(1#-ty)*yf1+ty*yf2     /* それに対応する転送「元」のy座標を求める
140     for xt=xt1 to xt2                          /* 転送「先」の各x座標について
150       tx=(xt-xt1)/dxt: xf=(1#-tx)*xf1+tx*xf2/* それに対応する転送「元」のx座標を求める
160       pset( xt, yt, point( xf, yf ) )
170     next /* xt
180   next /* yt
190 endfunc
```

リスト3

```
 10 /*   リスト3
 20 screen 1,3,1,1
 30 img_load( "test.gl3", 0, 0, 0 )
 40 move3( 0, 0, 199, 99,  0, 200, 299, 349 )  /* 拡大(縦横とも1.5倍)
 50 move3( 0, 0, 199, 99,  0, 400, 149, 474 )  /* 縮小(縦横とも0.75倍)
 60 end
 70 /* グラフィック拡大縮小ルーチン(Bresenhamアルゴリズム版)
 80 func move3( xf1;int, yf1;int, xf2;int, yf2;int, xt1;int, yt1;int, xt2;int, yt2;int )
 90   int xf, yf, xt, yt                         /* 実数型の変数の宣言がなくなった
100   int dxf, dyf, dxt, dyt, ex, ey             /* 代わりに誤差(error)の宣言が加わる
110   dxf=xf2-xf1: dyf=yf2-yf1
120   dxt=xt2-xt1: dyt=yt2-yt1
130   yf=yf1: ey=2*dyf-dyt                       /* 転送元の座標と誤差の初期値(y座標)
140   for yt=yt1 to yt2
150     xf=xf1: ex=2*dxf-dxt                     /* 転送元の座標と誤差の初期値(x座標)
160     for xt=xt1 to xt2
170       pset( xt, yt, point( xf, yf ) )
180       while ( ex>=0 )                        /* x方向の誤差が正なら
190         xf=xf+1                              /* 転送先のx座標を増やし
200         ex=ex-2*dxt                          /* x方向の誤差を減らす
210       endwhile
220       ex=ex+2*dxf                            /* x方向の誤差を増やす
230     next /* xt
240     while ( ey>=0 )                          /* y方向の誤差が正なら
250       yf=yf+1                                /* 転送先のy座標を増やし
260       ey=ey-2*dyt                            /* y方向の誤差を減らす
270     endwhile
280     ey=ey+2*dyf                              /* y方向の誤差を増やす
290   next /* yt
300 endfunc
```

▶大学3年から4年にもなるとシューショクジョーホーシが玄関に山になる毎日です。僕ってそんなに有名なのかな。なんてバカ書いてるときじゃない。どこから洩れるんでしょうね，住所や名前が。会ったこともないヤツから手紙もらって何がうれしいんだっていうの！(ホンネ)
山口　忠(21)X68000ACE-HD　岐阜県

ておいて，2回目以降はそれを参照するだけにする。これはきっと速くなる。というわけで秘密兵器「配列」に登場願おう。

dim int Xf(512)

X-BASICでは“dim”は省略可能らしいが，配列であることを誇示するためにつけておいた。

処理は2ステップに分かれる。第1ステップは前処理で，Bresenhamアルゴリズムを使って，ループ変数x_tに対応するx_fの値を計算し，いったん配列Xf(xt)に格納する。第2ステップは本処理である。y_fについては従来どおりのやり方で計算するが，x_fは配列から引いてくる。それがリスト4である。

さてリスト2とのリターンマッチの結果は？　転送する領域のサイズにもよるが，僕が試したところではリスト3に比べて実に4倍もの高速化となり，リスト2に比べても2倍以上速いという好結果が出た。整数化アルゴリズムの面目は保たれた。

余談になるが，y_fの配列は用意しなかった。その最大の理由は1回しか計算しないからである。そして，配列のアクセスというものは，単純変数のアクセスよりも時間がかかるものである。こういうことも知っておくとなかなか便利である。

point(xf, yf)

point(Xf(xt), Yf(yt))

見るからに下のほうが時間がかかりそうであろう。それでもx_fのほうには配列を用意したのは，計算の繰り返しを避けるメリットのほうがはるかに大きかったからである。くどいようだが，ここいらの見極めがなかなか微妙なのである。試しにy_fも配列で持つプログラムも作ってみたが，かえってリスト4より遅くなった。ヤミクモに配列を使えばいいというものでもないわけだ。

まだ少し改良の余地がなくもないが，これ以上の改良はプログラムが汚くなるので，このへんで打ち止めということにする。高級言語のプログラムは綺麗に書きたいから，これ以上は蛇足。

今後の課題

ここで演習問題を出しておこう。今回紹介したいくつかの転送ルーチンは，転送元の領域1と転送先の領域2が重なっていないときは正常に動くが，重なっていると異常な動作をすることもある。それはどんなときで，またそれを防ぐためにはどうすればいいか考えてほしい。x方向およびy方向のループの組み方にヒントがある。

もうひとつ演習のネタを。本ルーチンを使えば，違う画面モードにあうように転送することも可能である。利用例として，512×512ドットモードの画像を切り取って，768×512ドットモードで表示できるようにするというのがある。単純に転送すると，転送した画像が縦長になるので，転送先のx方向のドット数（xt2－xt1）を1.5倍にするだけでよい。

ただ，ファイルにいったん格納するなどの処置をしておかないと，screen命令を実行するとグラフィック画面が消えてしまうので要注意。さらに応用になるが，65536色モードから16色モードへ色数を落とす処理をかませれば，フルカラー画像がwidth96のモードで見られないといった悩みも解決である。色数を減らす方法は，たとえば1988年11月号で薬野氏が考案されたフルカラーを白黒に落とす記事を参照するといいであろう。必要に迫られて書いてみたが，プログラムは長くなるし，卑怯にもCで書いてしまったのでここには載せられない。これも，あくまで演習課題というところ。

＊　＊　＊

X-BASICは，関数のモジュール化や構造化プログラミングがしやすい仕様になっている。今回掲げた数本のプログラムはなるべくそうした特徴が生かせるように書いてある。Cに比べて少し甘いかなと思える部分もあるが，インタプリタとしてはいいセンいっていると思う。

今回のネタは，結果がすぐわかる，変な動作をしたらすぐわかる，という点で教材としての使い勝手がよいものと自負している。数年前僕がBASICを修得したのも，すべて画面まわりの扱いからだった。そのせいかいまだに音楽関係は不得手だが。

リスト4

```
10 /*   リスト4
20 screen 1,3,1,1
30 img_load( "test.gl3", 0, 0, 0 )
40 move4( 0, 0, 199, 99,  0, 200, 299, 349 )  /* 拡大(縦横とも1.5倍)
50 move4( 0, 0, 199, 99,  0, 400, 149, 474 )  /* 縮小(縦横とも0.75倍)
60 end
70 /* グラフィック拡大縮小ルーチン(Bresenhamアルゴリズム・配列導入版)
80 func move4( xf1;int, yf1;int, xf2;int, yf2;int, xt1;int, yt1;int, xt2;int, yt2;int )
90   int xf, yf, xt, yt
100   int dxf, dyf, dxt, dyt, ex, ey
110   dim int Xf(511)                          /* 転送元の座標を格納する配列はx座標だけ
120   dxf=xf2-xf1: dyf=yf2-yf1                 /* このあたりは転送の本体ではなく、
130   dxt=xt2-xt1: dyt=yt2-yt1                 /* 座標をあらかじめ計算しておく部分
140   xf=xf1: ex=2*dxf-dxt
150   for xt=xt1 to xt2
160     Xf(xt)=xf                              /* ここで配列に登録
170     while ( ex>=0 )
180       xf=xf+1
190       ex=ex-2*dxt
200     endwhile
210     ex=ex+2*dxf
220   next /* xt
230   yf=yf1: ey=2*dyf-dyt                     /* y座標の計算は転送しながら行う
240   for yt=yt1 to yt2                        /* 無駄に計算することはない
250     for xt=xt1 to xt2
260       pset( xt, yt, point( Xf(xt), yf ) )
270     next /* xt
280     while ( ey>=0 )
290       yf=yf+1
300       ey=ey-2*dyt
310     endwhile
320     ey=ey+2*dyf
330   next /* yt
340 endfunc
```


▶今年京都の八坂神社へ初詣に行った人は恐ろしい光景をみたであろう。あの境内左右に山積みされたメッコール400缶（だったと思う）の迫力は，並ではなかった。
箕浦　真（19）なし　岐阜県

BASICで作るXCオプティマイザ

プロトタイピングのすすめ

Nakamori Akira
中森　章

「プロトタイピング」というと難しそうですが，要は雛形を作り少しずつ必要な部分を加えてプログラムを完成させていく手法です。サンプルとして，簡単なXCのオプティマイザを作ってみましょう。プログラム設計の過程も参考にしてください。

プログラミングの動機

私たちはなんのためにプログラミングをするのでしょう。答えは簡単。作りたいものがあるからです。こんなことがいいな，できたらいいなと思う発想がプログラミングの第一歩なのです。これは現状に対する不満といっていいかもしれません。プログラミングにおいて必要なことは，文法を知っていることなんかではなくて自分がなにをしたいのかをはっきり認識することです。

私がいまいちばん不満に思っていることは，XCの性能が悪いということなのです。そこで，今回のテーマはXCのオプティマイザ（最適化を行うプログラム）です。ただし，ここではBASICのプログラムそのものについて語るつもりはありません。私がいいたいのはプログラミングのやり方，あるいはプログラミングをする場合になにを考えるべきかということです。

いわゆるソフトハウスでは，プログラミングは仕様書に基づいて行われます。つまり，仕様を決め，フローチャートを書き，それからプログラミングです。しかし個人でプログラムを作る場合，仕様書を書いたりフローチャートを書いたりするのは非常に煩わしいことです。その途中で挫折してしまう人も多いでしょう。実際，こんな一般論を守っていたのではプログラムが完成することはまずありません（多くの場合は納期があるので無理矢理完成させる）。

私自身は仕様なんてものは，他人にプログラムを作らせるためのものであって，個人でプログラミングする場合は不要なものだと思っています。仕様検討に長い時間を費やすよりも実際に動かしてみることのほうが大切です。つまり最初に簡単なプログラムを作り，動きを確認しながら元のプログラムを少しずつ拡張していくのです。この方法だと実際に動くプログラムがいつもあるのでプログラムを作ったという実感がありますし，少しずつ拡張することによる効果をすぐに確かめることができます。

このようなプログラミング方法はプロトタイプ（原型のこと，プロトカルチャーとは無関係）を作りながらプログラミングすることからプロトタイピングと呼ばれています。

さて，プロトタイピングを行う場合，

　　作成→実行→デバッグ

という動作を何回も繰り返すことになります。この点BASICなどのインタプリタ言語はすぐに修正できて実行できるという意味で最適です（コンパイラ言語を使っても構わないのですが）。今回はX-BASICを使ってプログラミングを行いますが，X-BASICを使うと上で述べた点のほかにも次のような利点があります。

●致命的なエラーに関してはX-BASIC自身がエラー処理をしてくれるので，エラー処理のためのプログラムが不要になってプログラムがすっきりする。

●実行速度の点で不満があれば，プログラム完成後にC言語のプログラムに変換して高速に実行することができる。

●文字列の処理に関しては（おそらく）ほかのどの言語よりも融通性がある。

それでは，XCのオプティマイザをプロトタイピングによって作成していく過程をレポートしていきたいと思います。

オプティマイザの位置づけ

まずは，どのようにXCの出力コードを最適化するか考えましょう。これはプログラム以前に考えておかなければならないことです。

ご存じのようにXCではコンパイル時にCCP（プリプロセッサ），CC0（構文解析），CC1（コード生成），CC2（オプティマイザ），AS（アセンブラ），LK（リンカ）というプログラムが順次実行されます。XCでコンパイルされたプログラムの性能が悪いのはCC1およびCC2から出力されるコードが悪いからだといえます。

そこで今回作成するオプティマイザはCC1やCC2とASのあいだに実行することでアセンブリ言語によるソースプログラムの改良を行うものとします。これから作るオプティマイザはアセンブリ言語のプログラムをよりよく変換するものなのです。

通常のコンパイルではCCというドライバがCCPからLKまでの処理を一気にやってしまいますから，新たなオプティマイザを挿入するためには少し細工が必要です。

まず/Sオプションによってコンパイルをアセンブラの直前で中断します。たとえば，TEST.Cというプログラムをコンパイルするのであれば，

　　CC　/S　/O　TEST.C

を実行します。するとTEST.Cをコンパイルして作られたTEST.S（拡張子が.Cから.Sに変わったもの）というアセンブリ言語のプログラムが作成されます。このTEST.Sというプログラムを今回作成するオプティマイザに通して別のファイル（たとえばTESTO.S）に変換します。

そして，このプログラムをもう一度CCでコンパイル（正確にはアセンブル）してやるのです。CCは拡張子が.Sのファイルについてはアセンブル以降の処理を実行しますから，たとえば，

　　CC TESTO.S

を実行すれば，実行形式のTESTO.Xというファイルが作成されます。これは，TEST.Cをコンパイルしてできたプログラムと同じ

▶シャープワープロ「書院 WD-A610」を買いました。やっぱり専用機はいいです。「将軍」もそれなりにいい機能はあるのですが，スピードがいけません。スカッとした満足感が得られることが，精神衛生上大切ですね。　渡辺　真澄（31）X1turbo，MZ-700　岐阜県

動きをするもの（しかし，さらに最適化されている）だとわかりますね。

まずは小手調べ

これまでの説明でわかるように，目的のプログラムはアセンブリ言語のプログラム（XCによって出力されるやつ）から別のアセンブリ言語への変換を行うプログラム（もちろん最適化をする）です。そこで，最初に作るべきプロトタイプはファイルを読み込んで別のファイルに書き出す（コピーする）のみのプログラムを作ってみましょう。これがオプティマイザのバージョン0.0です。ファイルをそのまま別のファイルにコピーするだけでもなんらかの変換（なにもしないという変換）をしたことに違いありませんからね。

具体的なプログラムはどうなるでしょう。入力するファイルを1行ずつ処理することにすれば，1行読んで1行書くという処理を入力ファイルの終わりまで繰り返せばいいことになります。つまり，ファイルをオープン（fopen）したあとは，読み込み（freads）と書き込み（fwrites）を入力ファイルの終わり（freadsの値が−1）まで繰り返し，最後にファイルをクローズ（fcloseall）するだけです。

これらの処理にどのような命令を使うかはX-BASICマニュアルの索引でファイル入出力のところを見ればいいでしょう。

オプティマイザのバージョン0.0のプログラムリストはリスト1のようになります。ただし，リスト1では入力ファイルの内容をファイルに書き出すだけでなくディスプレイ画面にも書いています。これは結果をすぐに見るためです。

どういう機能が必要か

ファイルが自由に読み込めるようになりましたから，最適化を行ううえで先々どのような機能が必要になるかを考えましょう。アセンブリ言語のプログラムの1行はラベル，命令，オペランドといくつかのフィールドに分かれていますから，おそらくそれらが別々に切り分けられていれば便利だと考えられます。これを考えましょう。

アセンブリ言語でプログラムを書いた人ならわかると思いますが，ラベル，命令，オペランドは空白やタブを区切りとして記述されています。そこでこの区切りを目印にフィールド分けを行うことができます。いま入力ファイルの1行は文字列変数に格納されていますから，その文字列変数の中で区切りの位置がわかればmid$関数（BASICでは非常によく使われる関数）によって，各フィールドを切り取ることができるのです。

入力ファイルの1行をラベル，命令，オペランドの各フィールドに分けてプリントするものをオプティマイザのバージョン0.1としましょう。今度は少し複雑です。

効率的な処理を行うためにいくつかの関数（またはサブルーチン）が必要になりそうです。プロトタイピングでは難しそうな処理を行う関数（できるだけ小規模なやつ）を最初に作り，それを組み合わせてより大きなプログラムを作ります。ここでは次のような関数が必要になると考えられます。

●skip_white：文字列変数の指定した文字位置から調べて初めて空白でもタブでもない文字がある位置を値とする関数

●get_token：逆に，文字列変数の指定した文字位置から調べて初めて空白かタブがある位置を値とする関数

これらの関数があれば，与えられた文字列変数に対してskip_white関数で返ってくる位置とget_token関数で返ってくる位置との間をmid$関数で取り出せばひとつのフィールドを得ることができます。

これを最大3回（ラベルがない場合は2回）繰り返せばラベル，命令，オペランドの各フィールドを得ることができますね。なおラベルがあるかないかは最初のフィールドの最後の文字が：かどうかを調べればよいでしょう。上の2つの関数ができればバージョン0.1は完成したも同然ですね。

以上のような方針で作ったプログラムがリスト2です。リスト2ではget_token関数による文字のサーチが文字列変数の最後で終了するようにstrlen関数で求めた文字列の長さを引数で与えていますが，get_tokenでは空白とタブ（文字コード9）のほかに文字列の終了（文字コード0）も調べているので冗長といえば冗長です。

なお，リスト2では文字列変数に［ ］をつけると［ ］内で指定する位置にある文字を参照できるという裏技（？）を使っています。この記述をしてもBCによって正常にX-BASICからCへの変換はできますから安心して使ってください。

また，できればskip_whiteやget_tokenなどという関数は作らずにX-BASICの組み込み関数を使いたかったのですが，ぴったりのものはありませんでした（strchrやstrcspnが近いのですが）。

リスト2の実行結果を写真1に示します。これはリスト3に示すアセンブリ言語のプログラムをオプティマイザの入力とするときの画面出力です。

リスト1

```
1000 /*+-----------------------------------------------------------------
1010 /*|  XC用 コード最適化プログラム への試み
1020 /*|
1030 /*|        バージョン0.0                              March 12, 1990
1040 /*|                                           programmed by 中森 章
1050 /*+---------------- メインプログラム -----------------------------------
1060 str LIN[256]
1070 int FP,FP2
1080 str infile,outfile
1090 input "入力ファイル"; infile
1100 input "出力ファイル"; outfile
1110 FP =fopen(infile,"r")
1120 FP2=fopen(outfile,"c")
1130 while 1
1140    if(freads(LIN,FP)=(-1)) then break /* EOF ならおしまい
1150    print LIN                          /* 画面に出力
1160    fwrites(LIN+chr$(13)+chr$(10),FP2)   /* ファイルに出力
1170 endwhile
1180 fcloseall()
1190 end
```

まだまだ必要な機能

さらにどんな機能が必要でしょう。通常オプティマイズは（C言語のプログラムでの）関数単位に行われます。このため，アセンブリ言語のプログラムの中で，どこからどこまでがひとつの関数をコンパイルしたものであるかを知ることが必要です。XCによって出力されるコードを眺めると関数の最初は¯¯で始まる，

¯¯関数名

▶近所というほど近くではないが，オール50円のゲームセンターができたので，金のあるうちにせっせと行き，上海2にハマっている毎日だ。
前田 賢一（18）X1turboZ 静岡県

として通常のラベルと区別しているのがわかります。したがってラベルの最初の2文字を調べればどこが関数の始まりかわかります。それでは関数の終わりはどうやって知ればいいのでしょう。このためには確実な方法はありませんが，経験的に.DC,.L以外の疑似命令，すなわち，.GLOBL,.COMM,.TEXT,.DATAなどが現れたら関数の終わりと思ってよいでしょう。

関数の始まりと終わりがわかれば，そのあいだに入力ファイルから読み込んだ各行はバージョン0.1と同様なフィールド分けを行って配列に保存しておきます。そしてこの配列をあとで作るオプティマイズ関数に渡して最適化を行うことになります。ここまでくればあとはオプティマイズ関数を作るだけ（！）のことになります。

この関数の始まりと終わりを認識するプログラムをバージョン0.2としましょう。なお，バージョン0.2ではバージョン0.1とは異なり，ラベルだけの行をできるだけ少なくして配列の効率的な利用をしています。すなわちラベルがある行には命令やオペランドは存在しない（XCの性格）ので，次の行に命令が記述されている場合（当然ラベルはない）は2つの行をひとつにして配列に格納しているのです。

バージョン0.2のプログラムをリスト4に示します。リスト4はかなり最終的なものに近いプログラムになっています。プログラムはさらに複雑になってしまいましたが，バージョン0.0，0.1と眺めてきた人は主な変更部分のみに着目すればよいのです。結局は次のような処理をしているというのがわかるでしょう。

すなわち，関数の外にあると判断される行に関してはそのまま画面（や出力ファイル）に書き出します（put_codeなんて関数を作ってありますね）。一方，関数内と判断される（関数の始まりから終わりのあいだにある）行はラベル，命令，オペランドをそれぞれFLABEL, OPC, FOPRNDという配列に格納していき（配列の上限はFPTRという変数に入っている），関数の終わりでoptimizeという関数を呼んで最適化処理（ここではopt_0〜6の7段階の処理が可能）を行い，それから画面（や出力ファイル）に書き出しているのです。

写真1　リスト2の実行結果

ただ，現在optimizeという関数の中で最適化処理はなにもしていません。下請けの関数であるopt_0〜6の実体を作ることで最適化処理ができるようになります。

ところで，リスト4では最適化レベルというものを入力するようにしています。これはopt_0からopt_6までの最適化処理のうち，どこまでの処理を行うかを指定するものです（差し当たってはあまり意味はありませんが）。なおリスト4で定義してあるshiftという関数はFLABEL, OPC, FOPRNDという配列間で要素の移動を行うものです。これはopt_0からopt_6までの処理を記述するときに必要となると思います。

リスト4を最適化レベル6で実行した結果を写真2に示します。これは写真1と同じリスト3に示すアセンブリ言語のプログラムをオプティマイザの入力とするときの画面出力です。ちゃんと関数ごとに分けて処理されているのがわかるでしょう。

リスト2

```
1000 /*+-----------------------------------------------------------------
1010 /*|  XC用  コード最適化プログラム  への試み
1020 /*|
1030 /*|        バージョン0.1                          March 12, 1990
1040 /*|                                          programmed by 中森  章
1050 /*+----------------- メインプログラム --------------------------------
1060 str LIN[256],FLABEL,OPC,FOPRND
1070 int FP,FP2,FPTR=-1,LNUM=0,bgn,fin,mrk
1080 str infile,outfile
1090 input "入力ファイル"; infile
1100 input "出力ファイル"; outfile
1110 FP =fopen(infile,"r")
1120 FP2=fopen(outfile,"c")
1130 while 1
1140  if(freads(LIN,FP)=(-1)) then break                 /* EOF ならおしまい
1150  LNUM=LNUM+1                                        /* 行番号を計算
1160  fin=strlen(LIN) : bgn=skip_white(0) : mrk=get_token(bgn+1,fin)
1170  if(LIN[mrk-1]=':') then {                          /* ラベルだった
1180         FLABEL=mid$(LIN,bgn+1,mrk-bgn-1)            /* : は含まない
1190         bgn=skip_white(mrk) : mrk=get_token(bgn+1,fin)
1200  } else FLABEL="<NONE>"
1210  OPC=mid$(LIN,bgn+1,mrk-bgn)                        /* 命令
1220  bgn=skip_white(mrk) : mrk=get_token(bgn+1,fin)
1230  FOPRND=mid$(LIN,bgn+1,mrk-bgn)                     /* オペランド
1240  LIN="ラベル="+FLABEL+chr$(9)+"命令="+OPC+chr$(9)+"オペランド="+FOPRND
1250  print LIN                                          /* 画面に出力
1260  fwrites(LIN+chr$(13)+chr$(10),FP2)                 /* ファイルに出力
1270 endwhile
1280 fcloseall()
1290 end
1300 /*------------------------------------------------------------------
1310 func skip_white(s;int)      /** スペース、タブをスキップする
1320   int i : i=s
1330   while(LIN[i]=' ' or LIN[i]=9) : i=i+1 : endwhile
1340   return (i)
1350 endfunc
1360 /*------------------------------------------------------------------
1370 func get_token(s;int,e;int) /** オペコード、ラベルなどを得る
1380   int i
1390   for i=s to e
1400     switch LIN[i]
1410     case ' ': return (i)       /* スペース
1420     case  9 : return (i)       /* タブ
1430     case  0 : return (i)       /* 改行
1440     endswitch
1450   next
1460   return (i)
1470 endfunc
```

リスト3

```
          include fefunc.h
          .COMM   _a,4
          .XREF   __main
          .XDEF   _main
          .TEXT
_main:
~~_main:
          BRA     L1
L2:
          MOVE.L  -4(A6),-(SP)
          JSR     _f
          ADDQ.L  #4,SP
          MOVE.L  D0,_a
L3:
          UNLK    A6
          RTS
L1:
          LINK    A6,#-4
          BRA     L2
          .GLOBL  _f
_f:
~~_f:
          BRA     L4
L5:
          MOVE.L  8(A6),D0
          ADD.L   #10,D0
          UNLK    A6
          RTS
L4:
          LINK    A6,#0
          BRA     L5
          .DATA
          .EVEN
          .END
```

▶MSX2を買おうとしたが，友人がX1turboZ IIを買ったのに刺激され，長年欲しかった「カラーイメージボード」と「試験に出るX1」を買ってしまった。気づいたときには貯金もなくなっていた。
増田　和通（16）X1F　静岡県

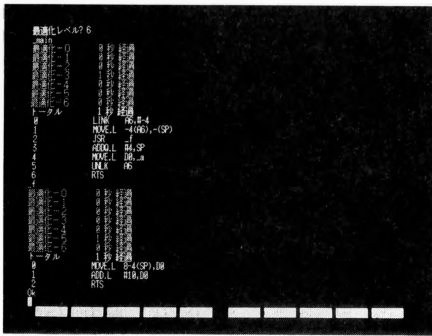



写真2　リスト4の実行結果

徐々にパワーアップ

バージョン0.2のプログラムができたらオプティマイザは完成したも同然です。あとはopt_0からopt_6までの関数を好みに合わせて作るだけです。また，opt_7，opt_8，……と最適化処理の段階を増やすことも容易でしょう。ですから，プログラムの説明は本来ならここで打ち切ってもよいのですが，それではあまりにひどいので具体的な最適化処理プログラムを示します。

リスト5がopt_0からopt_6までの最適化処理関数の具体例です。これは1日1関数の割合で私がねちねちと作ってきた（これも一種のプロトタイピングといえるかもしれない）もので，特にドライストンベンチマークを高速にするための最適化処理です(opt_7，opt_8と拡張されていく予定)。

ただし，リスト5ではopt_0からopt_6までの本体しか定義していませんから，実際に使用するためにはリスト4のプログラムと結合して使用する必要があります。

なおリスト5のプログラムは今回の原稿で本質的な部分ではないので説明はごく簡単にしようと思います。興味のある人は頑張って解読してみてください。

●最適化―opt_0

特定の命令をより高速な別の命令に置き換えています。ここでは，

CLR.L D0 → MOVEQ.L #0.D0
MOVEM.L Dn,-(SP) → MOVE.L Dn,-(SP)

という置き換えをしているのみです。それほど著しい効果は見られませんでした。

●最適化―opt_1

乗除算を高速化します。具体的にはファンクションコールである，

FPACK __CLMUL
FPACK __CLDIV

に関して，D0とD1をスタックに積むことで引数の受け渡しを行っている場合は，

FPACK __LMUL
FPACK __LDIV

というファンクションコール(直接D0とD1の値を引数として演算する）に変更して引数の受け渡しのためのスタック操作を省略します。

●最適化―opt_2

定数倍の乗算を高速化します。定数倍の場合はシフトと加算だけで乗算を行い，わざわざファンクションコールしないようにします。ただ引数がどのレジスタにあるか知るのが難しいのでファンクションコール__LMUL（D0とD1による引数渡し）における定数倍（D1が定数の場合）のみを対象とします。したがって，opt_1の処理をしたあとでないとopt_2は無意味になります(__LMULはXCでは出力しない)。

実際のコードは筆算で乗算を行う場合と同じ要領で，定数倍をシフトと加算を用いる命令列に展開していきます。もっと効率よい展開方法もあるのですが，ここではもっとも単純な方法を用いています。

●最適化―opt_3

XCでは関数の先頭で行うべきLINK処理(引数とローカル変数のベースアドレスの固定）は，次のように関数の後部に先頭から分岐して行われています。

```
_func:
        BRA    L0
L1:
        ……………
L0:
        LINK   A6, ＃数字
        ……………
        BRA    L1
```

これは見た目がよくありませんし，2個の無条件分岐（BRA）はまったくの無駄です。そこで，上の命令列を整理して，

```
_func:
        ……………
        LINK   A6, ＃*
        ……………
```

というように変換します。ただし，むやみに変換することはできませんからある条件に合致するときのみ変換しています。

また，opt_3では参照されないラベルを削除することもやっています。

●最適化―opt_4

ここでは不要な代入を削除したり，メモリをレジスタに置き換えることで高速化を行います。ここでは次の3つのケースを考慮しています。A，Bは適当なオペランド，reg，memはそれぞれレジスタ，メモリを示しています。

リスト4

```
1000 /*+------------------------------------------------------------
1010 /*|  XC用 コード最適化プログラム への試み
1020 /*|
1030 /*|        バージョン0.2                         March 12, 1990
1040 /*|                                         programmed by 中森 章
1050 /*+--------------- メインプログラム -----------------------------
1060 int MAXLIN=512
1070 str LIN[256],FNAME,FLABEL(512),OPC0,OPC(512),FOPRND(512)
1080 int FP,FP2,FPTR=-1,LNUM=0,bgn,fin,mrk
1090 str infile,outfile,solev : int olev=0
1100 input "入力ファイル"; infile
1110 input "出力ファイル"; outfile
1120 input "最適化レベル";solev
1130 if (solev="") then olev=0 else olev=val(solev)
1140 FP =fopen(infile,"r")
1150 FP2=fopen(outfile,"c")
1160 while 1
1170  if(freads(LIN,FP)=(-1)) then break /* EOF ならおしまい
1180  LNUM=LNUM+1
1190  fin=strlen(LIN) : bgn=skip_white(0) : mrk=get_token(bgn+1,fin)
1200  if(LIN[mrk-1]=':') then {  /* ラベル
1210     if(LIN[bgn]='¯')and(LIN[bgn+1]='¯') then { /* 関数の最初
1220        FNAME=mid$(LIN,bgn+3,mrk-bgn-3) : FPTR=0 : FLABEL(0)=""
1230        put_code(LIN)
1240     } else if(FPTR=-1) then { /* 関数外のラベル
1250        put_code(LIN)
1260     } else { /* 関数内のラベル
1270        if(FLABEL(FPTR)<>"") then { /* 連続するラベル
1280           OPC(FPTR)="" : FOPRND(FPTR)="" : FPTR=FPTR+1
1290           if(FPTR>MAXLIN) then abort("関数が長すぎる")
1300        }
1310        FLABEL(FPTR)=mid$(LIN,bgn+1,mrk-bgn-1)  /* ラベルを入れる
1320     }                                          /* (: は含まない)
1330  } else { /* ラベルでない
1340     OPC0=mid$(LIN,bgn+1,mrk-bgn)
1350     if(term_func(OPC0)and(FPTR<>-1)) then { /* 関数の終わりかも?
1360        optimize() : FPTR=-1                 /* 終わりなら最適化する
1370        put_code(LIN)                        /* 終わりを見付けた行を書く
1380     } else if (FPTR=-1) then { /* 関数外
1390        put_code(LIN)                        /* 読んだ行をそのまま書く
```

▶花粉症に悩まされる方もいると思いますが(実は僕もそうです)，頑張ってください。
柏木　克久（14）X68000PRO，MSX2　静岡県

ケース1：無意味な代入

　MOVE.L　A,B

　MOVE.L　B,A

という命令列を，

　MOVE.L　A,B

に変換します。

ケース2：メモリリードをレジスタへ

　MOVE.L　reg,mem

　MOVE.L　mem,B

という命令列を，

　MOVE.L　reg,B

　MOVE.L　reg,mem

に変換します。

ケース3：中間的なレジスタの削除

　MOVE.L　A,reg

　MOVE.L　reg,B

という命令列があるとき，次にregが変更されるまでregの値が使われていないなら，

　MOVE.L　A,B

に変換します。

●最適化—opt_5

ある条件を満たす場合，LINK処理とそれに対応するUNLK処理を削除します。

- 関数内でスタックの変化はない
- LINK命令/UNLK命令以外にA6を使用してない

という2条件を満たす場合は単純にLINK命令とUNLK命令を削除します。ただし，A6を使用している場合も，d(A6)のかたち(ディスプレースメントつき)でしか現れない場合は，

　d(A6) → d−4(SP)

という置き換え（ディスプレースメントから4を引く）を行ったあと，LINK命令とUNLK命令を削除します。

●最適化—opt_6

ループ命令（DBRA）でループ回数が定数で与えられている場合は，ループを行わずその処理をループ回数だけ書き並べます。たとえば，

　MOVE.L　#2,D0

L0：

　MOVE.W　(A1)+,(A0)+

　DBRA　D0,L0

という命令列は，

　MOVE.W　(A1)+,(A0)+

　MOVE.W　(A1)+,(A0)+

　MOVE.W　(A1)+,(A0)+

と展開されます。また，ここでは，

```
1400     } else { /* 関数内
1410        OPC(FPTR)=OPC0                        /* 命令を入れる
1420        bgn=skip_white(mrk) : mrk=get_token(bgn+1,fin)
1430        FOPRND(FPTR)=mid$(LIN,bgn+1,mrk-bgn+1)  /* オペランドを入れる
1440        FPTR=FPTR+1 : if(FPTR>MAXLIN) then abort("関数が長すぎる")
1450        FLABEL(FPTR)=""                       /* 次の行のための初期化
1460     }
1470  }
1480 endwhile
1490 fcloseall()
1500 end
1510 /*------------------------------------------------------------
1520 func skip_white(s;int)      /** スペース、タブをスキップする
1530   int i : i=s
1540   while(LIN[i]=' ' or LIN[i]=9) : i=i+1 : endwhile
1550   return (i)
1560 endfunc
1570 /*------------------------------------------------------------
1580 func get_token(s;int,e;int) /** オペコード、ラベルなどを得る
1590   int i
1600   for i=s to e
1610     switch LIN[i]
1620     case ' ': return (i)      /* スペース
1630     case  9 : return (i)      /* タブ
1640     case  0 : return (i)      /* 改行
1650     endswitch
1660   next
1670   return (i)
1680 endfunc
1690 /*------------------------------------------------------------
1700 func put_code(s;str)        /* 文字列をプリントする
1710   fwrites(s+chr$(13)+chr$(10),FP2)
1720 endfunc
1730 /*------------------------------------------------------------
1740 func abort(s;str)           /* アボート処理
1750   print "Line:";LNUM,FNAME,s
1760   fcloseall() : end
1770 endfunc
1780 /*------------------------------------------------------------
1790 func term_func(s;str)       /* 関数の終わりを知る
1800   if(s="include") then return (1)
1810   if(s[0]<>'.')   then return (0)
1820   if(s=".GLOBL")  then return (1)
1830   if(s=".COMM")   then return (1)
1840   if(s=".TEXT")   then return (1)
1850   if(s=".DATA")   then return (1)
1860   if(s=".XDEF")   then return (1)
1870   if(s=".XREF")   then return (1)
1880   if(s=".END")    then return (1)
1890   return(0)
1900 endfunc
2000 /*------------------------------------------------------------
2010 func shift(i;int,j;int)    /* テーブルの内容をシフトする
2020   FLABEL(i)=FLABEL(j) : OPC(i)=OPC(j) : FOPRND(i)=FOPRND(j)
2030 endfunc
2040 /*------------------------------------------------------------
2050 func optimize()              /* 最適化処理
2060   int t : str st
2070   color(1) : print FNAME : color(2)                   /* 関数名を表示
2080                 print "最適化－0", :st=time$:opt_0():t=ptime(st)
2090   if (olev>0) then print "最適化－1", :st=time$:opt_1():t=ptime(st)+t
2100   if (olev>1) then print "最適化－2", :st=time$:opt_2():t=ptime(st)+t
2110   if (olev>2) then print "最適化－3", :st=time$:opt_3():t=ptime(st)+t
2120   if (olev>3) then print "最適化－4", :st=time$:opt_4():t=ptime(st)+t
2130   if (olev>4) then print "最適化－5", :st=time$:opt_5():t=ptime(st)+t
2140   if (olev>5) then print "最適化－6", :st=time$:opt_6():t=ptime(st)+t
2150   color(3) : print "トータル",t;"秒 経過"
2160   for i=0 to FPTR-1                    /* 結果を画面表示
2170     print i,FLABEL(i),OPC(i),FOPRND(i)
2180     if(FLABEL(i)<>"") then put_code(FLABEL(i)+":")
2190     put_code(chr$(9)+OPC(i)+chr$(9)+FOPRND(i))
2200   next
2210 endfunc
3000 /*------------------------------------------------------------
3010 func opt_0()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
3020 endfunc
4000 /*------------------------------------------------------------
4010 func opt_1()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
4020 endfunc
5000 /*------------------------------------------------------------
5010 func opt_2()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
5020 endfunc
6000 /*------------------------------------------------------------
6010 func opt_3()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
6020 endfunc
7000 /*------------------------------------------------------------
7010 func opt_4()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
7020 endfunc
8000 /*------------------------------------------------------------
8010 func opt_5()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
8020 endfunc
9000 /*------------------------------------------------------------
9010 func opt_6()                 /* 将来の拡張用　お好きな最適化処理をどうぞ
9020 endfunc
9900 /*------------------------------------------------------------
9910 func int ptime(s;str)                           /* 経過時間を表示
9920   int st,et : str e : e=time$
9930      st=val(mid$(s,1,2))*3600+val(mid$(s,4,2))*60+val(mid$(s,7,2))
9940      et=val(mid$(e,1,2))*3600+val(mid$(e,4,2))*60+val(mid$(e,7,2))
9950      print (et-st);"秒 経過"
9960      return (et-st)
9970 endfunc
```

▶名古屋商科大学経営情報学科に合格しました。この大学では「学生全員へのパソコン無償譲渡制度」とかいうことで，IBMのパソコン「PS/55モデル5530Z」なるものが家に届いた。　　豊田　貴統（18）MZ-1500，PS/55モデル5530Z　静岡県

MOVE.W　(A1)+,(A0)+

が2個続く場合は,

MOVE.L　(A1)+,(A0)+

に置き換える処理もやっています。

●最適化の効果

それぞれの最適化の効果を見るためにドライストンベンチマークによる比較を行ってみましょう。表1がOPMDRVをOFFにしたときのドライストンベンチマーク(バージョン2.1)の結果です。GCCでの結果には及ぶべくもありませんが,opt_0からopt_6までのすべての最適化を行う場合はXCでそのままコンパイルしたものに対して6割の性能向上をすることができました。大健闘といえるのではないでしょうか。

あとはローカル変数をレジスタに割りつけるなどの最適化を行うと倍に近い性能を得ることも夢ではなくなるでしょう。

おわりに

本当はもっと短いプログラムを作るつもりだったのですが,少しずつ改造していくうちに巨大なものになってしまいました。まあ,これがプロトタイピングの醍醐味といえるかもしれません。こんなプログラムを仕様から考えていたのではプログラム規模を考えただけで作る気がなくなってしまいますからね。皆さんもプロトタイピングによってもっと気楽にプログラミングしてみてはいかがでしょうか。パソコンでゲームばかりやっていては体験できない新鮮な感動を得ることができるでしょう。

ところで,世間には初心者はC言語と混乱してしまうのでX-BASICを使わないほうがいいという意味不明の論理が横行していますが,これは英語と混同するからドイツ語を覚えてはいけないといっているようなものです。気にしないでどんどんプログラミングしましょう。

表1　最適化の効果

最適化	ドライストン値	プログラムサイズ
オリジナル	568	9658 バイト
レベル 0	568	9646 バイト
レベル 1	617	9580 バイト
レベル 2	877	9620 バイト
レベル 3	892	9590 バイト
レベル 4	909	9574 バイト
レベル 5	909	9556 バイト
レベル 6	925	9590 バイト

リスト5

```
1000 /*+-----------------------------------------------------------------
1010 /*|  XC用 コード最適化プログラム への試み
1020 /*|
1030 /*|        バージョン0.3                          March 12, 1990
1040 /*|                                           programmed by 中森 章
1050 /*+----------------- メインプログラム --------------------------------
3000 /*-------------------------------------------------------------------
3010 func opt_0()          /* (0) 単純な置換
3020   int i : str s,t
3030   for i=0 to (FPTR-1)
3040     s=FOPRND(i) : t=s
3050     if((OPC(i)="CLR.L")and(s[0]='D')) then {
3060        OPC(i)="MOVEQ.L" : FOPRND(i)="#0,"+FOPRND(i)
3070     }
3080     if(OPC(i)="MOVEM.L") then {
3090        s[1]='0'
3100        if((s="D0,-(SP)")or(s="A0,-(SP)")) then OPC(i)="MOVE.L"
3110        t[7]='0'
3120        if((t="(SP)+,D0")or(t="(SP)+,A0")) then OPC(i)="MOVE.L"
3130     }
3140   next
3150 endfunc
4000 /*---------------------------------------------------------------------
4010 func opt_1()          /* (1) 引数のレジスタ渡し
4020   int p,i
4030   p=FPTR-1
4040   repeat
4050     if((OPC(p)="FPACK")and((FOPRND(p)="__CLMUL")or(FOPRND(p)="__CLDIV"))) then {
4060        if((OPC(p-1)="MOVEM.L")and(FOPRND(p-1)="D0/D1,-(SP)")) then {
4070           OPC(p-1)="FPACK" : FOPRND(p-1)="__"+right$(FOPRND(p),4)
4080           for i=(p+2) to (FPTR-1) : shift(i-2,i) : next : p=p-1 : FPTR=FPTR-2
4090        }
4100     }
4110     p=p-1
4120   until (p<0)
4130 endfunc
5000 /*----------------------------------------------------------------------
5010 func opt_2()          /* (2) 定数倍の展開
5020   int p,x,i : str s : p=FPTR-1
5030   repeat
5040     if((OPC(p)="FPACK")and(FOPRND(p)="__LMUL")and(OPC(p-1)="MOVE.L")) then {
5050        s=FOPRND(p-1)
5060        if(s[0]='#') then {
5070           x=strchr(s,',')
5080           if(mid$(s,x+2,2)="D1") then {
5090              x=val(mid$(s,2,x))
5100              if(x=0) then {
5110                 OPC(p-1)="MOVEQ.L" : FOPRND(p-1)="#0,D0"
5120                 for i=(p+1)to(FPTR-1) : shift(i-1,i) : next : p=p-1 : FPTR=FPTR-1
5130              } else if (x=1) then {
5140                 for i=(p+1)to(FPTR-1) : shift(i-2,i) : next : p=p-1 : FPTR=FPTR-2
5150              } else if (x>1)and(x<=256) then {
5160                 const_mult(x,p) : p=p-1
5170              }
5180           }
5190        }
5200     }
5210     p=p-1
5220   until (p<1)
5230 endfunc
5240 /**
5250 func const_mult(x;int,p;int)    /* 定数倍の本体
5260   str topc(16),toprnd(16)
5270   int i,j,m,pre,aft,mv,tp
5280   mv=0 : tp=0 : m=&H100
```


▶1カ月,パソコンをやってなかったら,ブラインドタイプができなくなってしまった。
杉森　孝治'(18) X68000EXPERT-HD 愛知県

```
5290   for i=0 to 8
5300      if(x and m)<>0 then break
5310      m=m shr 1
5320   next : pre=8-i
5330   m=m shr 1
5340   while m<>0
5350     for i=0 to (pre-1)
5360        if(x and m)<>0 then break
5370        m=m shr 1
5380     next : aft=pre-i-1
5390     if(m=0) then {
5400         topc(tp)="ASL.L" : toprnd(tp)="#"+str$(pre)+",D0"
5410         tp=tp+1 : break
5420     } else {
5430         if(mv=0) then {
5440            topc(tp)="MOVE.L" : toprnd(tp)="D0,D1"
5450            tp=tp+1 : mv=1
5460         }
5470         topc(tp  )="ASL.L" : toprnd(tp  )="#"+str$(pre-aft)+",D0"
5480         topc(tp+1)="ADD.L" : toprnd(tp+1)="D1,D0"
5490         tp=tp+2
5500     }
5510     m=m shr 1 : pre=aft
5520   endwhile
5530   if(tp>2) then {
5540     for i=(p+1)to(FPTR-1) : j=FPTR-i+p : shift(j+tp-2,j) : next
5550   }else if (tp<2) then {
5560     for i=(p+1)to(FPTR-1) : shift(i+tp-2,i) : next
5570   }
5580   FPTR=FPTR+tp-2
5590   for i=0 to (tp-1)
5600      FLABEL(p+i-1)="" : OPC(p+i-1)=topc(i) : FOPRND(p+i-1)=toprnd(i)
5610   next
5620 endfunc
6000 /*------------------------------------------------------------------------
6010 func opt_3()          /* (3) 関数のプロローグ処理を先頭に
6020   int i,j,k,kk,eb : str L1,L2
6030   if (OPC(0)<>"BRA") then remo_label() : return ()
6040   L2=FLABEL(1) : if (L2="")  then remo_label() : return ()
6050   L1=FOPRND(0)
6060   for i=1 to (FPTR-1)
6070      if (FLABEL(i)=L1) then break    /* 絶対に見つかるハズ
6080   next
6090   if (OPC(i-1)<>"RTS") then remo_label() : return ()
6100   for j=i to (FPTR-1)
6110      if (OPC(j)="BRA") then break
6120   next
6130   if (j<>(FPTR-1))   then eb=0 else eb=1 /* BRA は最終行か
6140   if ((FOPRND(j)<>L2)or(FLABEL(j)<>"")) then return ()
6150   for k=1 to (FPTR-1) : kk=FPTR-k : shift(kk+(j-i-1),kk) : next
6160   for k=0 to (j-i-1) : shift(k,k+i+(j-i-1)) :next
6170   if (eb=0) then {
6180     for k=(j+1+(j-i-1))to(FPTR-1+(j-i-1)) : shift(k-(j-i+1),k) : next
6190   }
6200   FPTR=FPTR-2
6210   remo_label()
6220 endfunc
6230 func remo_label()                            /* 不要なラベルの削除
6240   int i
6250   for i=0 to (FPTR-1)
6260      if ((FLABEL(i)<>"")and(is_used(FLABEL(i))=0)) then FLABEL(i)=""
6270   next
6280 endfunc
6290 func int is_used(x;str)       /* ラベルが参照されているかどうかを調べる
6300   int i
6310   for i=0 to (FPTR-1)
6320      if (instr(1,FOPRND(i),x)<>0) then return(1)
6330   next
6340   return(0)
6350 endfunc
7000 /*------------------------------------------------------------------------
7010 func opt_4()         /* (4)不要な レジスタへの代入を削除
7020   int i,j,p,q : str S0,S1,D0,D1 : i=0
7030   while (i<FPTR)
7040      if ((OPC(i)="MOVE.L")and(OPC(i+1)="MOVE.L")and(FLABEL(i+1)="")) then {
7050         p=strchr(FOPRND(i),',')+1 : q=strchr(FOPRND(i+1),',')+1
7060         D0=mid$(FOPRND(i)  ,p+1,32) : S0=left$(FOPRND(i)  ,p-1)
7070         D1=mid$(FOPRND(i+1),q+1,32) : S1=left$(FOPRND(i+1),q-1)
7080         if (D0<>S1)          then i=i+1 : continue
7090         if (S0=D1) then {  /* A->B ; B->A
7100            for j=(i+2) to (FPTR-1) : shift(j-1,j) :next : FPTR=FPTR-1
7110            continue
7120         }
7130         if (((S0[0]='D')or(S0[0]='A'))and(D0[0]<>'D')) then {
7140            FOPRND(i+1)=S0+","+D1 : i=i+1 : continue /* A->mem ; mem->B
7150         }
7160         if (alive(i+2,D0)=1) then i=i+1 : continue
7170         FOPRND(i)=S0+","+D1                         /* A->reg ; reg-> B
7180         for j=(i+2) to (FPTR-1) : shift(j-1,j) : next : FPTR=FPTR-1
7190      }
7200      i=i+1
7210   endwhile
7220 endfunc
7230 func int alive(p;int,r;str)   /* レジスタを壊してよいかを調べる
7240   int i,q : str S,D
7250   for i=p to (FPTR-1)
7260     S=OPC(i)
7270     if (S="RTS")       then return (0) /* 最後まで使われない
7280     if (S="JMP")       then return (1) /* 分岐があるとわからない
7290     if (S="JSR")       then {          /* 関数内では D0-D2 は変更される
7300        if((r="D0")or(r="D1")or(r="D2"))then return(0) else return(1)
```

▶いままでは，Oh！P○やOh！F○と並べてあると違和感があったOh！Xだがこれから
3冊並んでいても違和感がなくなった。

今川　健久（15）X68000ACE，MZ-2000　愛知県

```
7310      }
7320      if (S[0]='B')       then {           /* 条件分岐があるとお手上げ
7330         if(S="BRA") then i=search_lab(FOPRND(i))-1 : continue
7340         return (1)
7350      }
7360      if (instr(1,FOPRND(i),r)<>0) then { /* レジスタを使用?
7370         if (OPC(i)<>"MOVE.L") then return (1) /* 使用している可能性あり
7380         q=strchr(FOPRND(i),',')+1
7390         D=mid$(FOPRND(i),q+1,32) : S=left$(FOPRND(i),q-1)
7400         if (instr(1,S,r)<>0) then return (1) /* ソースに使用
7410         if (D=r)              then return (0) /* 使われていない
7420      }
7430    next
7440    return(0)  /* まったく使われていない
7450 endfunc
7460 func search_lab(x;str)     /* ラベルをサーチする
7470    int i
7480    for i=0 to (FPTR-1)
7490       if(FLABEL(i)=x) then return (i)
7500    next
7510    return (-1)   /* ここに来ることはないハズ
7520 endfunc
8000 /*---------------------------------------------------------------------
8010 func opt_5()          /* (5) LINK / UNLK を省略してみる
8020    int i,j,r=0 : str op
8030    for i=0 to (FPTR-1)
8040      if (OPC(i)="LINK") then continue
8050      if (OPC(i)="UNLK") then continue
8060      if (instr(1,FOPRND(i),"A6")<>0) then {
8070        if(instr(1,FOPRND(i),"(A6)")=0) then return()
8080        r=1
8090      }
8100    next
8110    if (r=1) then {
8120       for i=0 to (FPTR-1)  /* SP の変更がないかチェック
8130         op=FOPRND(i)
8140         if (instr(1,op,",SP")<>0)   then return()
8150         if (instr(1,op,"-(SP)")<>0) then return()
8160         if (instr(1,op,"(SP)+")<>0) then return()
8170         if (OPC(i)="JSR")           then return()
8180    next
8190    }
8200    i=0
8210    while i<FPTR
8220      if ((OPC(i)="LINK")or(OPC(i)="UNLK")) then {
8230         if (FLABEL(i+1)<>"") then {
8240            OPC(i)="" : FOPRND(i)=""
8250         } else {
8260            FLABEL(i+1)=FLABEL(i)
8270            for j=(i+1) to (FPTR-1) : shift(j-1,j) : next : FPTR=FPTR-1
8280         }
8290      }
8300      i=i+1
8310    endwhile
8320    if (r=0) then  return ()
8330    for i=0 to (FPTR-1)
8340      op=FOPRND(i)
8350      for j=0 to 1           /* オペランドは最大2個
8360        r=instr(1,op,"(A6)")
8370        if (r<>0) then {
8380           op=left$(op,r-1)+"-4(SP)"+mid$(op,r+4,32)
8390        } else break
8400      next : FOPRND(i)=op
8410    next
8420 endfunc
9000 /*---------------------------------------------------------------------
9010 func opt_6()          /* (6) DBRA を展開する
9020    int i,j,k,p: str tmp,cnt,lbl,opc,opr
9030    for i=0 to (FPTR-1)
9040       if (OPC(i)="DBRA") then {
9050          p=strchr(FOPRND(i),',')+1
9060          cnt=left$(FOPRND(i),p-1) : lbl=mid$(FOPRND(i),p+1,32)
9070          if (FLABEL(i-1)<>lbl)   then return ()
9080          if (alive(i+1,cnt)=1)   then return ()
9090          if (OPC(i-2)<>"MOVE.L") then return ()
9100          p=strchr(FOPRND(i-2),',')+1
9110          if (mid$(FOPRND(i-2),p+1,32)<>cnt) then return ()
9120          tmp=left$(FOPRND(i-2),p-1) : if (tmp[0]<>'#') then return ()
9130          p=val(mid$(tmp,2,32))+1
9140          opc=OPC(i-1) : opr=FOPRND(i-1)
9150          if (p<4) then {
9160            for j=(i+1)to(FPTR-1) : shift(j-(3-p),j) : next
9170          } else {
9180            for j=(i+1)to(FPTR-1) : k=FPTR+i-j : shift(k+(p-3),k) : next
9190          }
9200          FPTR=FPTR+(p-3)
9210          for j=1 to p
9220             FLABEL(i+j-3)="" : OPC(i+j-3)=opc : FOPRND(i+j-3)=opr :next
9230       }
9240    next
9250    for i=0 to (FPTR-1)
9260       if ((OPC(i)="MOVE.W")and(FOPRND(i)="(A1)+,(A0)+")) then {
9270         if (FLABEL(i+1)<>"") then continue
9280         if ((OPC(i+1)="MOVE.W")and(FOPRND(i+1)="(A1)+,(A0)+")) then {
9290            FLABEL(i+1)=FLABEL(i)
9300            for j=(i+1) to (FPTR-1) : shift(j-1,j) : next : FPTR=FPTR-1
9310         }
9320         OPC(i)="MOVE.L"
9330       }
9340    next
9350 endfunc
```


▶あらあら「トドの大和煮」。毒物飲料のお次は「ベルリンの壁」ですか。
今井田　和也（17）X68000ACE-HD　愛知県

THE SENTINEL

●スタック型言語へのアプローチ

S-OSの他機種への移植が再び始まりました。CPUの違いを乗り越えFM-7にも移植されたS-OSの今回のターゲットはなんと16ビットマシンです。かなり早い時期にお届けすることができるのではないかと思います。ご期待ください。

さて，magiFORTH以来スタック型言語はながらく掲載されていませんでした。C言語が開発の主流となり，親分であるFORTH自体が最近ではあまり流行らなくなっているという理由もあるのかもしれません。

スタック型言語はスタックを基本とした演算や処理を行います。ほかのプログラミング言語で1+2は，「1 2 +」となります。これはまずスタックに1を積み，続いて2を積み，スタックに積まれている2つの数を加えて答えをまたスタックに積むという動作をします。プログラムを日本語的に読めるという特長を持っていて，先の例は「1と2を足す」と読むことができます。

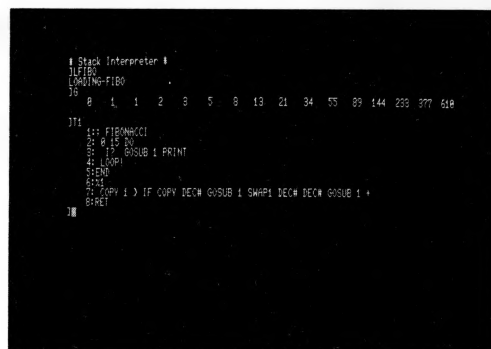
この特長を最大限に発揮する言語としてMINDがあります。PC-9800シリーズ用にパブリックドメインソフトとしてPUBLIC MINDが配布されていますので目にした方もあるでしょう。123という数を「123個」のようにも書ける面白い言語です。

第92部
インタプリタ言語STACK

●STACKという言語

MINDはFORTHの日本語的アプローチですが，今回のSTACKはFORTHのBASIC的アプローチといえるでしょう。FORTHは新しいコマンド(ワードと呼ばれる)を自分で作り出すことによってシステムを拡張しながら，目的のプログラムを作っていくという自己増殖型の言語です。新しく定義したワードはその場でコンパイルされ，最初から用意されているワードと区別なく使うことができるようになります。

STACKはこの機能を切り離し，BASICのように組み込みのワードを使ってプログラムを作り，それを実行するという方法を採用しました。BASICの表記方法を逆ポーランド記法にしたようなものだと思えば，プログラミングもそれほど難しくないでしょう。

スタック型言語はそもそも構造が簡単で，比較的容易に処理系を作ることができます。CやBASICなどの言語では必要不可欠な式の評価順序の決定（掛け算は足し算より先に計算しなければならないなど）の必要もありません。FORTHから自動コンパイル機能とワード管理機能を取り除いたことで処理系はコンパクトに収まっています。

●S-OSの系譜(10)

華々しく1周年を迎えたS-OS“SWORD”に新しい仲間が登場しました。1987年7月号でSMC-777用の“SWORD”が発表されたのです。SMC-777はソニーの8ビットコンピュータで，国産機としては珍しくASCIIキーボードを使っていました。多くのマシンはJISキーボードを使っていますが，記号の位置が海外のマシンとは異なっています。またSony FilerというDOSを標準装備しており，CP/Mが使えました。

S-OSとしての機能とCP/Mマシンとしての機能を両立させたため，このSMC-777版“SWORD”は強力なものとなりました。S-OSからCP/Mのファイルを読み書きできるようになっていたのです。

また，7月号では好評のmagiFORTHの機能アップが果たされました。これによりmagiFORTHで32ビット整数が扱えるようになり，計算力が大幅にアップしたのです。±21億まで扱うことのできる32ビット整数なら，大抵の計算はまにあいます。この変更によりmagiFORTHはより実用的な言語になったといえるでしょう。

続く8月号ではMZ-2500用の“SWORD”が発表されました。これまでは2500をMZ-2000モードにし，さらにMZ-2000用のBASICを入手して初めて“SWORD”が実行できたわけです。実行できる“SWORD”はMZ-2000用のものですから，MZ-2500の16ドットフォント表示もできず，高速なMZ-2000（MZ-2500は6MHzのMZ-2000として使うモードがあった）という環境に甘んじるしかなかったユーザーには非常に嬉しい移植となりました。

同時に発表となった「対局五目並べ」は，コンピュータとの対局を再現してくれる面白いものでした。この五目並べは連珠のルールに則ったもので，しかも定石を採用していないのにめっぽう強く，編集室中のMZ/X1がすべて五目並べマシンになるほどの盛況ぶりでした。

1986年9月号では当時史上最強のBASICと評価されたFuzzyBASICが発表されます。これは来月紹介しましょう。

全機種共通S-OS"SWORD"要

インタプリタ言語STACK

平井氏によるオリジナル言語シリーズも第4弾になります。今回はFORTHにちょっと似た逆ポーランド記法によるインタプリタ言語です。しかも，命令の並べ方以外はBASICと同じような仕様を持つという変わり種。なかなか面白い試みです。

Hirai Shinji
平井 真二

STACKはS-OS"SWORD"上で動作する，4バイト型整数およびMAGICをサポートした簡易エディタつきのインタプリタ言語です。

パラメータの受け渡しをスタックを通して行う点でかなりFORTHに似ていますが，元々は逆ポーランド記法のBASICもどきを作ろうという発想で生まれた言語です。そのため，GOTOやGOSUB～RETURNがあり，逆にFORTHの特徴でもあるワードを作るという機能はありません。

スタック領域を除いて8Kバイト弱の小さな言語ですが，機能面でも速度でもFuzzy BASICには劣りません。セミコンパイラ使用時にはさらに1.5～2倍ほど速くなります。

入力&実行方法

使用するプログラムはリスト1の1本だけです。MACINTO-Cなどのマシン語入力ツールから打ち込んで，実行アドレス3000Hでセーブしてください。

S-OS"SWORD"のモニタから，

　#J3000

と入力すると，コールドスタートします。また，3003Hにジャンプすればホットスタート（テキストエリアを初期化しない）します。

タイトル表示後，プロンプト']'が表示され，入力待ちになります。]に続いてコマンド一覧のコマンドを入力するとそのコマンドが実行されます。また，1文字以上のスペースをつけてSTACKの命令を入力すると，その命令が実行されます（ダイレクトモード)。

例)

　]　5　BELL

ビープ音が5回鳴ります。

エディタを使ううえでの注意点は，

1)　テキストの入力において，I（追加)，B(挿入）を使い分けなければならない。

2)　nはラベルでなく，Tコマンドのリスト表示でエディタがつけた行番号である。

3)　nは省略できない（Dコマンドの第2パラメータを除く)。

です。なお，コマンド中断にはシフト＋ブレイクしてください。プログラムの実行はGコマンドです。

実行中エラーが出ると，エラーメッセージとエラーが出た行を表示して止まります。エラーコード表を参照して修正してください。また，シフト＋ブレイクを押すと，プログラムの実行を中断できます。

インタプリタでもそこそこのスピードはありますが，速度が要求される場合はセミコンパイラを使ってみてください。使い方は，テキストおよびスタックと重ならないアドレスを求めて，

　］Cアドレス

と入力してください。実行は，

　］Jアドレス

です（アドレスはCコマンドで入力した値と同じ)。

セミコンパイラ使用時は，エラーチェックをほとんど行わないので，エラーが出ないことが確認されたテキストを用いてください。また，Jコマンドで実行したプログラムはBREAK文があるところ以外はシフト＋ブレイクはききません。

表1　エディタコマンド一覧

コマンド	機能
I	テキストエンドより追加入力を始める。シフト＋ブレイクでコマンド入力に戻る
Tn	n行からテキストを表示する。スペースで一時停止，シフト＋ブレイクでコマンド入力に戻る。なお，表示後のテキストはスクリーンエディット可能
Dn1, n2	n1からn2行を削除する(,n2は省略可)
Bn	n行からテキストの挿入を始める。シフト＋ブレイクでコマンド入力に戻る
Sファイル名	現在作成しているテキストをファイル名でセーブする
Lファイル名	ファイル名のテキストをロードする
Z	ディレクトリを表示する
&	現在作成中のテキストを消去する
R	テキストを復活する
P	Tコマンドにおけるプリンタ出力のON/OFFを設定する。デフォルトはOFF
F文字列	先頭から文字列を探し始める。スペースで一時停止，シフト＋ブレイクでコマンド入力に戻る
M	現在作成中のテキストの格納されているアドレスを表示する
Xアドレス	テキスト格納先頭アドレスを指定する。デフォルトは4E00H番地。なお，アドレスは16進4桁で，また，初めて指定したときは必ず&を実行すること
!	S-OSに戻る
G	プログラムを実行する
Cアドレス	テキストをセミコンパイルする。結果はアドレス以降に格納される
Jアドレス	セミコンパイルしたプログラムを実行する。アドレスはCコマンドで指定したアドレスである

図1　メモリマップ

アドレス	内容
3000H	テキストエディタ
33D3H	インタプリタ本体
4684H	ワークエリア
49A7H	ラベルエリア
4DA6H	
4E00H	テキストエリア ↓ ／ ↑ パラメータスタック
AA00H	リターンスタック
AE00H	

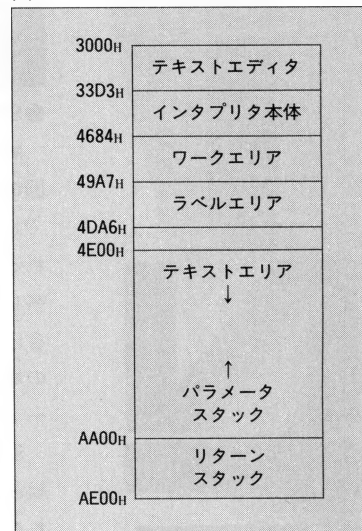



表2　エラーメッセージ

SYNTAX ERROR
　ワードの記述がおかしい
STACK EMPTY
　パラメータスタックが空になった
RETURN STACK EMPTY
　リターンスタックが空になった
UNDEFINED LABEL
　分岐先のラベルが見あたらない
OUT OF LABEL
　ラベルが2047を超している

▶ X68000のローンで毎日バイトの嵐。テスト週間でバイトを休んでも，ドラクエⅣをやってちゃ，しょーがねえなあ。あと半年のローン，誰かなんとかして～。あぁ赤点が怖い。
松野　竜明（17）X68000PRO，X1C　愛知県

プログラムについて

ソースリストを見ればわかりますが，ワードの判別は，頭の１文字と文字列の ASCIIコードの和の下位1バイトでチェックしています。この方法は，テーブルが小さくてすみ，プログラムも簡単ですが，ワード名を決めるのに苦労します。一応，ワードが追加できるようにはなってますが上記の理由により面倒ですのでわかる人はやってみてください。

ラベルは1～511まではテーブルを用いているので速いですが，512以上はテキストの先頭からサーチするので遅いです。プログラムを書くときには気をつけてください。

セミコンパイラの処理は，ワードの解析，定数とラベルの数値化，変数の格納アドレスのオフセットアドレスを求めるだけですので，実行レベルはインタプリタと同等です。

スタックは空かどうかしかチェックしてません（速度優先）。

最後に

サンプルプログラムを見てもわかるように，FORTHとBASICを足して２で割ったような感じの言語になりました。STACKでプログラムを書く際には，それらの言語を参考にするとよいでしょう。

文字列を使うには自分でメモリを管理しなければならないとか，負数を表現できるのに符号つきの乗除算ルーチンがない，ファイル関係の命令がないなどの欠点はありますが，プログラムサイズを考えるといい線いってると思います。

書き忘れてましたが，グラフィック関係のワードを使うには，あらかじめグラフィックパッケージ“MAGIC”をロードしておいてください。

今後の予定としては，ファイル関係の命令を追加する，フルコンパチのコンパイラを作るなどがありますが，仕事が忙しくなってきましたのでいつになるかはわかりませんが，できるだけ実現しようと思います。

Profile

◇平井さんは神奈川県にお住まいの21歳，会社員でX1turboユーザーです。TTI，TTCシリーズでもうすっかりお馴染みでしょう。

サンプルプログラム

```
1 ; FIBONACCI
2  0 15 DO
3   I?  GOSUB 1 PRINT
4  LOOP!
5 END
6 %1
7  COPY 1 > IF COPY DEC# GOSUB 1 SWAP1 DEC# DEC# GOSUB 1 +
8 RET
```

STACKリファレンスマニュアル

<>内はスタックの状態で「──」の左が動作直前，右が動作直後の状態を表す。この中でd_H，d_L，d，c，addはそれぞれ4バイトデータの上位２バイト，４バイトデータの下位２バイト，２バイトデータ，２バイトデータの下位バイト，addはアドレス（２バイト）を示す。

構文規則

- ラベルは1～2047の範囲で必要なところだけ行の先頭に%を続けて書く。
- セミコロン以下の１行は注釈とみなす。
- ワードおよび定数のあいだはスペースで区切る。
- 負数は２の補数表現。
- 文字列は$0D_H$または”を終端文字とする。

定数

10進数（４バイトデータ） <──d_H d_L>
- 先頭に’をつけて表す。とりうる値は’0～’4294967295。上位，下位の順番でスタックに積まれる。

10進数（２バイトデータ） <──d>
- とりうる値は0～65535。

16進数 <──d>
- 先頭に＄をつけて表す。とりうる値は$0000～$FFFF。

文字列 <──add>
- ダブルクォーテーションで囲まれた文字列の先頭アドレスをスタックに積む。

変数

- Ａ～Ｚおよびそれらに１～９をつけた260個が使える。
- ４バイトデータは続くもうひとつの変数が使われる。

例）Ａに４バイトデータを代入すると，Ａに上位２バイト，A1に下位２バイトが代入される。

出力用ワード

PRINT <d──>
- スタックトップを，右詰め10進５桁で出力する。

PRINT1 <d──>
- スタックトップを，左詰め10進で出力する。

PRINT2 <d_H d_L──>
- スタックトップの32ビット数を，左詰め10進で出力する。

PRF <d──>
- スタックトップを，左詰め符号つき10進で出力する。

PRF2 <d_H d_L──>
- スタックトップの32ビット数を，左詰め符号つき10進で出力する。

CHR <c──>
- スタックトップの下位バイトをASCIIコードとみなし，対応する文字を出力する。

CR <──>
- 復帰改行を出力する。

PRTS <add──>
- スタックトップのアドレスから$0D_H$または” の直前までの文字列を出力する。

COTR <add──>
- スタックトップのアドレスから $0D_H$ または”の直前までの文字列をコントロール文字列とみなして出力する。

カーソルコントロール文字列……画面制御
- D……カーソルを下へ１文字分移動
- U……カーソルを上へ１文字分移動
- R……カーソルを右へ１文字分移動
- L……カーソルを左へ１文字分移動
- C……画面をクリア
- ／……改行する

HEX2 <c──>
- スタックトップの下位バイトを16進２桁で出力する。

HEX4 <d──>
- スタックトップを16進４桁で出力する。

HEXL <d_H d_L──>
- スタックトップの32ビット数を16進８桁で出力する。

PRON <──>
- 画面出力をプリンタにも出力するようにする。S-OSの＃LPTONと同様。

PROFF <──>
- 画面出力をプリンタには出力しないようにする。S-OSの＃LPTOFF と同様。

入力用ワード

KEY <──c>
- キーが押されるのを待って１文字入力し，結果をスタックに積む。

GETKEY <──c>
- リアルタイムキー入力。キー入力のないときは０を返す。

FLGET <──c>
- カーソルを点滅させて１文字入力する。

INP$ <add──>
- キーボードから１行入力し，結果をadd以降に格納する。入力文字列の最後には$0D_H$が付加される。

演算用ワード

＋ <d1 d2──d3>
- d1とd2の和を求める（d3）。

－ <d1 d2──d3>

・d1とd2の差を求める（d3)。
＊　＜d1 d2——d3＞
・d1とd2の積を求める（d3)。
／　＜d1 d2——d3＞
・d1をd2で割った商を求める（d3)。
MOD　＜d1 d2——d3＞
・d1をd2で割った余りを求める（d3)。
/MOD　＜d1 d2——d3 d4＞
・d1をd2で割った商（d3）と余り（d4）を求める。
L＋　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{2H} d_{2L}＞
・スタック上の２つの32ビット数の和を32ビットで求めスタックに積む。
L－　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{2H} d_{2L}＞
・スタック上の２つの32ビット数の差を32ビットで求めスタックに積む。
L＊　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{2H} d_{2L}＞
・スタック上の２つの32ビット数の積を32ビットで求めスタックに積む。
L/　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{2H} d_{2L}＞
・スタック上の２つの32ビット数について２番目をトップで割り，その商を32ビットで求めスタックに積む。
LMOD　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{2H} d_{2L}＞
・スタック上の２つの32ビット数について２番目をトップで割り，その余りを32ビットで求めスタックに積む。
L/MD　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{2H} d_{2L} d_{3H} d_{3L}＞
・スタック上の２つの32ビット数について２番目をトップで割り，その商と余りをそれぞれ32ビットで求め，余り，商の順にスタックに積む。
D＊　＜d1 d2——d_{3H} d_{3L}＞
・d1とd2の積を32ビット数で求める。
＊！　＜d1 c——d2＞
・スタックトップを８ビット数とみなして積を求める。
D/MOD　＜d_{0H} d_{0L} d1——d3 d_{3H} d_{3L}＞
・スタックトップの16ビット数で２番目の32ビット数を割り，その商を32ビット，余りを16ビットで求め，余り，商の順にスタックに積む。
AND　＜d1 d2——d3＞
・d1とd2のANDをとる（d3)。
OR　＜d1 d2——d3＞
・d1とd2のORをとる（d3)。
XOR　＜d1 d2——d3＞
・d1とd2のXORをとる（d3)。
INC＃　＜d1——d2＞
・d1に１を加えスタックに積む（d2)。
DEC＃　＜d1——d2＞
・d1から１を引きスタックに積む（d2)。
HIGH　＜d1——d2＞
・d1の上位バイトを値としてスタックに積む（d2)。
LOW　＜d1——d2＞
・d1の下位バイトを値としてスタックに積む（d2)。
EX　＜d1——d2＞
・d1の上位バイトと下位バイトを交換したものを値としてスタックに積む(d2)。
NOT　＜d1——d2＞
・d1をビット反転する（d2)。
NEGATE　＜d1——d2＞
・d1の２の補数をとる（d2)。
ROR　＜d1 d2——d3＞
・d1の値をd2回右シフトしてスタックに積む（d3)。
ROL　＜d1 d2——d3＞
・d1の値をd2回左シフトしてスタックに積む（d3)。
CTL　＜d1——d_{2H} d_{2L}＞
・d1の値を符号つきで32ビット数に変換してスタックに積む。
＝＝　＜d1 d2——d3＞
・d1とd2の値が等しければ１を，等しくなければ０をスタックに積む。
！＝　＜d1 d2——d3＞
・d1とd2が等しくなければ１を，等しければ０をスタックに積む。
＜　＜d1 d2——d3＞
・d1＜d2ならば１を，そうでなければ０をスタックに積む。
＞　＜d1 d2——d3＞
・d1＞d2ならば１を，そうでなければ０をスタックに積む。
F＜　＜d1 d2——d3＞
・符号つき数値として処理する。動作は＜と同じ。
＝0　＜d1——d2＞
・d1が０と等しければ１を，そうでなければ０をスタックに積む。
CMP2　＜d_{1H} d_{1L} d_{2H} d_{2L}——d3＞
・２つの32ビット数を比較して，d1＞d2ならば１を，d1＝d2ならば０を，d1＜d2ならば－1(65535)をスタックに積む。

スタック操作用ワード

DROP　＜d——＞
・スタックトップ（d）を捨てる。
COPY　＜d——d d＞
・スタックトップ（d）をコピーしスタックに積む。
SWAP1　＜d1 d2——d2 d1＞
・スタックトップ（d2）と２番目（d1）を交換する。
ROT　＜d1 d2 d3——d2 d3 d1＞
・スタックの３番目（d1）がトップにくるように，スタックの上から３個のデータを回転する。
TR　＜d——＞
・スタックトップのデータ（d）をリターンスタックに積む。
FR　＜——d＞
・リターンスタックのトップをスタックに積む（リターンスタックのトップは取り去られている)。
DROPL　＜d_H d_L——＞
・スタック上の32ビット数を取り去る。
COPYL　＜d_H d_L——d_H d_L d_H d_L＞
・スタック上の32ビット数をコピーする。
SWAPD　＜d_{0H} d_{0L} d_{1H} d_{1L}——d_{1H} d_{1L} d_{0H} d_{0L}＞
・スタック上の２つの32ビット数を入れ替える。

機械語リンク用ワード

CALL　＜add——＞
・addの示すアドレスの機械語サブルーチンをコールする。
PUTA　＜c——＞
・スタックトップの値をＡレジスタにロードする。
PUTD　＜d——＞
・スタックトップの値をDEレジスタにロードする。
PUTH　＜d——＞
・スタックトップの値をHLレジスタにロードする。
GETA　＜——c＞
・Ａレジスタの値をスタックに積む。
GETD　＜——d＞
・DEレジスタの値をスタックに積む。
GETH　＜——d＞
・HLレジスタの値をスタックに積む。

メモリ操作用ワード

PEEKB　＜add——c＞
・addの示すアドレスからの１バイトデータをスタックに積む。
PEEKW　＜add——d＞
・addの示すアドレスからの２バイトデータをスタックに積む。
POKEB　＜c add——＞
・addの示すアドレスにｃを格納する。
POKEW　＜d add——＞
・addの示すアドレスに２バイトデータｄを格納する。
STRCPY　＜add1 add2——＞
・add1の示すアドレスから始まる文字列をadd2の示すアドレス以降にコピーする。
LEFT$　＜add1 add2 d——＞
・add1の示すアドレスから始まる文字列の左からｄ文字をadd2の示すアドレス以降に格納する。
RIGHT$　＜add1 add2 d——＞
・add1の示すアドレスから始まる文字列の右からｄ文字をadd2の示すアドレス以降に格納する。
MID$　＜add1 add2 d1 d2——＞
・add1の示すアドレスから始まる文字列の左からd1文字目以降のd2文字をadd2に示すアドレス以降に格納する。
STRCAT　＜add1 add2——＞
・add1の示すアドレスから始まる文字列に，add2の示すアドレスから始まる文字列を連結する。
STRLEN　＜add——d＞
・addの示すアドレスから始まる文字列の長さをスタックに積む。
INSTR　＜add c——d＞
・addの示すアドレスから始まる文字列中に，ｃが示す文字があればその位置を返し，みつからなければ０を返す。
STRCMP　＜add1 add2——d＞
・add1の示すアドレスから始まる文字列１とadd2の示すアドレスから始まる文字列２を比較し，一致したら０を，文字列１＞文字列２なら１を，文字列１＜文字列２ならー１を返す。
STRW　＜d add——＞

▶僕の友達のゲーマーが，X68000かPC-9801か，どちらを買おうかと迷っていたので，僕はMSXを買うことをすすめてしまいました。そして彼は本当にMSXを買ってしまいました。
安川　実（16）X1turboZ，MZ-2500　愛知県

・d の値を文字列に変換し，addの示すアドレス以降に格納する。

STRL <d_H d_L add――>

・32ビット数を文字列に変換し，addの示すアドレス以降に格納する。

VAL1 <add――d>

・addの示すアドレスから始まる10進文字列を数値化してスタックに積む。

VAL2 <add――d_H d_L>

・addの示すアドレスから始まる10進文字列を数値化（32ビット数）してスタックに積む。

VAL$ <add――d>

・addの示すアドレスから始まる16進文字列を数値化してスタックに積む。

TRANS＋ <add1 add2 d――>

・add1を転送元，add2を転送先として，d バイトをブロック転送する。機械語のLDIRと同等。

TRANS－ <add1 add2 d――>

・add1を転送元，add2を転送先として，d バイトをブロック転送する。機械語のLDDRと同等。

FILL <add d c――>

・addの示すアドレスから d バイトを c で埋める。

I/O操作用ワード

IN <add――c>

・addの示すアドレスのI/Oポートから入力された値をスタックに積む。

OUT <c add――>

・addの示すアドレスのI/Oポートに c を出力する。

特殊ワーク操作用ワード

PEEK＃ <add――c>

・addをオフセットアドレスとして，S-OS の特殊ワークエリアの 1 バイトデータを読み込みスタックに積む。

POKE＃ <c add――>

・addをオフセットアドレスとして，S-OS の特殊ワークエリアに c を書き込む。

一般ワード

WIDCH <c――>

・画面の桁数を指定する。

BELL <c――>

・c の回数だけビープ音を鳴らす。

LOCATE <d1 d2――>

・d1をＸ座標，d2をＹ座標とする位置へカーソルを移動する。

CURX <――d>

・カーソルのＸ座標をスタックに積む。

CURY <――d>

・カーソルのＹ座標をスタックに積む。

ASCII <add――d>

・addの示すアドレスから始まる文字列のASCIIコードをスタックに積む。

RND <――d>

・0～65535の範囲で乱数を発生する。

SCRN <d1 d2――c>

・d1をＸ座標，d2をＹ座標とするカーソル位置のキャラクタを読み出す。

＃ <――d>

＃変数名

・変数の値を 2 バイト単位で取り出しスタックに積む。

＃＃ <――d_H d_L>

＃＃変数名

・変数の値を 4 バイト単位で取り出しスタックに積む。

. <d――>

. 変数名

・スタックトップの値を変数に代入する。

.. <d_H d_L――>

..変数名

・スタックトップの32ビット数を変数に代入する。

INC <――>

INC 変数名

・変数の値に 1 を加える。

DEC <――>

DEC 変数名

・変数の値を 1 減ずる。

制御用ワード

GOTO

GOTO ラベル

・指定行へ分岐する。ワードとラベルのあいだはスペースで区切る。

GOSUB

GOSUB ラベル

・サブルーチンを呼び出す。

RET

・GOSUBに対応するRETURN。

IF <d――>

IF words

・スタックトップの値が 0 以外（要するに真）ならば words を実行する。BASICのIF文と同等。

REPEAT UNTIL

・「REPEAT words UNTIL」：wordsを実行し，UNTILの直前でスタックトップの値が真ならばループを脱し，そうでなければ再び繰り返す。

DO LOOP!

・「d1 d2 DO words LOOP!」として使用する。d1 は始値，d2は終値。wordsを繰り返す。

I?

・I？：ループカウンタの値をスタックに積む。

J?

・J？：ひとつ上のループカウンタの値をスタックに積む。

LEA

・LEA：ループカウンタの値を終値に一致させループ脱出の条件を満たす。

BREAK

・ブレイクキーが押されているかをチェックし，押されていればプログラムの実行を終了する。

END

・プログラムの実行を終了する。

グラフィック操作用ワード

INIT <――>

・画面を初期化する。以下のワードを使用する前に必ずこのワードを実行すること。

COL <d1 d2――>

・デフォルトのプレーンおよびモードを設定する。d1がプレーン，d2がモードを示す。内容は以下のとおり，

プレーン	モード
0……BLUE	0……RESET
1……RED	1……XOR
2……GREEN	2……OR
	3……NOP

PALET@ <c0 c1 c2 c3 c4 c5 c6 c7――>

・パレットを設定する（内容はMAGICと同じ)。

WIND <d1 d2 d3 d4――>

・(d1, d2)－(d3, d4)を結ぶ直線を対角線とする領域をグラフィックウィンドウに設定する。

CLS <c――>

・c で示されるプレーンを消去する。なお c が 3 のときは全プレーンを消去する。

DOT <d1 d2――>

・(d1, d2) に点を描画する。

LINE@ <d1 d2 d3 d4――>

・(d1, d2)－(d3, d4) 間を直線で結ぶ。

SLINE@ <d1 d2 d3 d4 d5 d6――>

・(d1, d2)－(d3, d4)－(d5, d6) の 3 点を滑らかに結ぶ曲線を描く。

BOX@ <d1 d2 d3 d4――>

・(d1, d2)－(d3, d4) を結ぶ直線を対角線とする長方形を描く。

TILE <d1 d2――>

・タイルパターンを設定する。

BOXFUL <d1 d2 d3 d4――>

・(d1, d2)－(d3, d4)を結ぶ直線を対角線とする長方形の領域をタイルパターンで塗りつぶす。

TRIANGLE <d1 d2 d3 d4 d5 d6――>

・(d1, d2)－(d3, d4)－(d5, d6)を頂点とする 3 角形の領域をタイルパターンで塗りつぶす。

CIRCLE <d1 d2 d3――>

・(d1, d2)を中心，d3を半径とする円をタイルパターンで塗りつぶす。

POINT <d1 d2――d>

・座標 (d1, d2) のパレットコードをスタックに積む。

MAGIC <add――>

・addで示されるアドレスから置かれたデータ列をMAGICに渡す。

※文字列関係のワードで転送先を指定するものがありますが，転送先がテキストにならないようにしてください。テキストを破壊しかねません。また，転送元と転送先のアドレスが重ならないようにしてください。

```
3000 C3 06 30 C3 31 30 CD E2 : CC
3008 1F 0C 2A 20 53 74 61 63 : 00
3010 6B 20 49 6E 74 65 72 70 : FD
3018 72 65 74 65 72 20 2A 00 : 6C
3020 2A C3 33 22 BF 33 22 C1 : 17
3028 33 36 00 21 00 00 22 C5 : 71
3030 33 ED 73 C8 33 CD EB 1F : 65
3038 3E 5D CD F4 1F ED 5B 76 : 39
3040 1F CD D3 1F 1A 13 FE 5D : 66
3048 C2 9E 31 1A 13 FE 49 CA : CF
3050 E3 31 FE 54 CA EA 31 FE : 49
3058 26 28 C5 FE 44 CA 1C 32 : 6D
3060 FE 42 CA 56 32 FE 53 CA : AD
3068 77 32 FE 52 28 4E FE 4C : B9
3070 CA B9 32 FE 58 28 52 FE : 83
3078 4D 28 5F FE 50 28 6E FE : B6
-------------------------------------
SUM: 03 F3 AA E4 B8 77 F9 39 DDE2

3080 21 CA FA 1F FE 47 CA E3 : F6
3088 33 FE 57 CA 12 31 FE 46 : D9
3090 CA 21 31 FE 5A CA 8B 31 : FA
3098 FE 43 CA 11 41 FE 4A CA : 6F
30A0 D3 33 FE 20 28 E5 CD E5 : D5
30A8 46 1A 13 23 77 B7 20 F9 : DD
30B0 36 0D 23 36 00 DD 21 E6 : 80
30B8 46 C3 E7 33 ED 5B BF 33 : 5D
30C0 3E 20 12 CD 5F 33 C3 31 : C3
30C8 30 CD B2 1F DA B5 33 22 : B2
30D0 C3 33 22 BF 33 CD 5F 33 : 69
30D8 18 EC 2A BF 33 CD BE 1F : CA
30E0 3E 2D CD F4 1F 2A C1 33 : 69
30E8 CD BE 1F 18 D9 CD EB 1F : 72
30F0 CD E2 1F 50 52 49 4E 54 : 5B
30F8 45 52 20 00 3A C7 33 EE : D9
-------------------------------------
SUM: 17 74 A2 6A 52 43 FE 54 6660

3100 01 32 C7 33 11 CC 33 B7 : F4
3108 20 03 11 CF 33 CD E5 1F : 07
3110 18 B4 3A 5C 1F FE 28 28 : CF
3118 02 3E 14 87 CD 30 20 18 : 10
3120 A5 DD 2A BF 33 01 00 00 : 9F
3128 ED 43 CD 46 DD 22 C8 46 : 59
3130 ED 4B CD 46 03 ED 43 CD : 4B
3138 46 CD C7 1F 31 30 DD 7E : B5
3140 00 DD 23 FE 0D 28 E5 B7 : CF
3148 CA 31 30 47 ED 5B 76 1F : 4F
3150 13 13 1A B8 20 E3 DD E5 : BD
3158 13 1A B7 28 0C DD 46 00 : 3B
3160 DD 23 B8 28 F3 DD E1 18 : A9
3168 D0 DD E1 2A CD 46 CD 0E : A6
3170 40 3E 3A CD F4 1F ED 5B : E0
3178 D1 46 CD E8 1F CD EB 1F : C2
-------------------------------------
SUM: AE 1E 75 7B 6D 59 55 02 1DD8

3180 DD 7E 00 DD 23 FE 0D 20 : 86
3188 F7 18 A1 CD 06 20 C3 31 : 97
3190 30 E5 CD 18 20 3E 06 85 : E3
3198 6F CD 1E 20 E1 C9 1B CD : 0C
31A0 7C 33 D5 01 01 00 CD FB : 4E
31A8 32 ED 5B 76 1F 13 13 13 : 48
31B0 13 13 1A FE 3A C2 B5 33 : 22
31B8 CD 91 31 13 1A CD 4F 33 : 0B
31C0 D1 CD 31 33 2A 76 1F 23 : E4
31C8 23 23 23 23 23 ED B0 ED : 39
31D0 5B 76 1F CD D3 1F 1A FE : C7
31D8 20 DA C6 30 FE 5D CA 31 : 46
31E0 30 18 BC ED 5B C1 33 C3 : 03
31E8 59 32 CD 7C 33 3A C7 33 : 3B
31F0 32 7C 1F D5 E5 CD 0E 40 : A2
31F8 E1 D1 3E 3A CD F4 1F CD : D7
-------------------------------------
SUM: 0C E3 26 35 FC 62 AF 59 369A

3200 E8 1F CD EB 1F AF 32 7C : 3B
3208 1F 1A FE 0D 13 20 FA 1A : 8B
3210 B7 CA 31 30 CD C7 1F 31 : C6
3218 30 23 18 D1 CD A4 33 DA : BA
3220 B5 33 DD 7E 00 FE 2C 20 : 8D
3228 21 DD 23 E5 CD A7 33 DA : 87
3230 B5 33 D1 ED 52 DA B5 33 : BA
3238 CA B5 33 EB 4B 42 03 C5 : F2
3240 CD 81 33 C1 CD FB 32 C3 : FF
3248 31 30 CD 81 33 01 01 00 : E4
3250 CD FB 32 C3 31 30 CD 7C : 67
3258 33 D5 ED 5B 76 1F CD D3 : 85
3260 1F 1A FE 1B 20 03 D1 18 : 5E
3268 EA CD 4F 33 D1 CD 31 33 : 3B
3270 2A 76 1F ED B0 18 E2 ED : 43
3278 4B BF 33 0A B7 CA B5 33 : B0
-------------------------------------
SUM: BF BB D6 D9 35 F8 FB 10 3623

3280 ED 43 70 1F 2A C1 33 B7 : 94
3288 ED 42 23 22 72 1F 21 00 : 26
3290 00 22 6E 1F 3E 04 CD A3 : 61
3298 1F CD AF 1F DA B2 33 CD : 46
32A0 EB 1F CD E2 1F 57 52 49 : CA
32A8 54 49 4E 47 2D 00 CD 9D : C9
32B0 1F CD AC 1F DA B2 33 18 : 8E
32B8 9A 3E 04 CD A3 1F CD 09 : 41
32C0 20 DA B2 33 CA D8 32 CD : 80
32C8 EB 1F CD E2 1F 53 4B 49 : BF
32D0 50 2D 00 CD 9D 1F 18 E6 : 04
32D8 CD EB 1F CD E2 1F 4C 4F : 40
32E0 41 44 49 4E 47 2D 00 CD : 5D
32E8 9D 1F 2A C1 33 22 70 1F : 8B
32F0 CD A6 1F DA B2 33 CD 5F : 7D
32F8 33 18 BC ED 43 CA 33 D5 : 09
-------------------------------------
SUM: F7 19 67 19 54 73 C4 99 53AB
```

```
3300 1A B7 CA B5 33 13 FE 0D : A1
3308 28 02 18 F4 0B 78 B1 20 : 8A
3310 EF 2A C1 33 B7 ED 52 4D : 50
3318 44 EB D1 03 ED B0 1B ED : A8
3320 53 C1 33 2A C5 33 ED 4B : A1
3328 CA 33 B7 ED 42 22 C5 33 : FD
3330 C9 C5 D5 2A C1 33 E5 09 : 6F
3338 22 C1 33 E5 B7 ED 52 4D : 3E
3340 44 D1 E1 ED B8 2A C5 33 : BD
3348 23 22 C5 33 D1 C1 C9 01 : 99
3350 01 00 FE 00 28 05 03 13 : 42
3358 1A 18 F7 3E 0D 12 C9 01 : 50
3360 00 00 ED 5B BF 33 1A FE : 52
3368 00 28 08 13 FE 0D 20 F6 : 64
3370 03 18 F3 ED 53 C1 33 ED : 2F
3378 43 C5 33 C9 CD A4 33 38 : E0
-------------------------------------
SUM: 45 58 1C 87 FC 44 FF 9C 6499

3380 34 ED 5B C5 33 EB B7 ED : 03
3388 52 EB 38 29 7C B5 28 25 : 1C
3390 E5 ED 5B BF 33 2B 7C B5 : 7B
3398 20 02 E1 C8 1A FE 0D 13 : 03
33A0 20 FA 18 F1 D5 DD E1 CD : 83
33A8 5D 3F CD 6B 3F 7C B5 C0 : 04
33B0 37 C9 CD 33 20 ED 7B C8 : 50
33B8 33 CD C4 1F C3 31 30 00 : 07
33C0 4E 00 4E 00 4E 00 00 00 : EA
33C8 00 00 00 00 4F 4E 00 4F : EC
33D0 46 46 00 CD B2 1F DA B5 : B9
33D8 33 22 E0 46 E5 DD E1 3E : 5C
33E0 01 18 05 DD 2A BF 33 AF : C6
33E8 ED 73 CB 46 32 DF 46 21 : E9
33F0 EE 49 11 EF 49 01 FF 03 : 83
33F8 36 00 ED B0 21 00 AE 22 : C4
-------------------------------------
SUM: 4B D2 41 F8 ED 29 8A 66 0E9D

3400 E2 46 21 00 AA F9 3A DF : 05
3408 46 B7 C2 E3 42 21 00 00 : 05
3410 39 01 01 AA B7 ED 42 D2 : 9D
3418 28 40 CD CD 1F CA 52 40 : 7D
3420 CD 5D 3F DD 7E 00 4F DD : F0
3428 23 B7 CA 07 41 FE 0D 28 : 1F
3430 D5 FE 40 30 67 FE 25 CA : 97
3438 D5 34 FE 3B CA EF 34 FE : 2D
3440 22 CA 01 35 FE 23 CA 07 : 14
3448 35 FE 2E CA 29 35 FE 24 : AB
3450 CA 7E 35 FE 2D 28 14 FE : E2
3458 27 28 25 FE 30 38 3D FE : 15
3460 3A 30 39 DD 2B CD 6B 3F : 22
3468 E5 18 9B DD 7E 00 FE 21 : 12
3470 38 09 CD 6B 3F CD EC 39 : AA
3478 E5 18 8B 06 2D C3 B1 34 : 63
-------------------------------------
SUM: A7 5B AD CF 4B D1 A2 B2 5A98

3480 DD 7E 00 FE 2D 28 08 CD : 83
3488 89 3F E5 D5 C3 06 34 DD : 5C
3490 23 CD 89 3F CD F4 39 E5 : 97
3498 D5 C3 06 34 47 DD 7E 00 : 74
34A0 DD 23 FE 20 28 0B FE 0D : 5C
34A8 28 07 FE 09 28 03 80 18 : F9
34B0 EB 78 21 19 44 85 30 01 : 97
34B8 24 6F 7E 5F 21 43 46 85 : 9F
34C0 30 01 24 6F 7E B9 C2 22 : DF
34C8 40 16 00 21 19 45 19 19 : 07
34D0 5E 23 56 EB E9 CD 6B 3F : 22
34D8 7C FE 02 30 0A 29 DD E5 : A1
34E0 C1 CD 0F 39 C3 06 34 FE : D1
34E8 08 D2 4C 40 C3 06 34 06 : 69
34F0 00 DD 7E 00 DD 23 FE 0D : 66
34F8 CA 06 34 B8 CA 06 34 18 : D8
-------------------------------------
SUM: 4F 18 98 C3 70 FE A4 C2 7570

3500 F0 DD E5 06 22 18 EA DD : B9
3508 7E 00 FE 23 28 0A CD 4B : E9
3510 35 5E 23 56 D5 C3 06 34 : DE
3518 DD 23 CD 4B 35 5E 23 56 : 24
3520 D5 23 5E 23 56 D5 C3 06 : 6D
3528 34 DD 7E 00 FE 2E 28 0A : ED
3530 CD 4B 35 D1 73 23 72 C3 : E9
3538 06 34 DD 23 CD 4B 35 D1 : 58
3540 C1 71 23 70 23 73 23 72 : F0
3548 C3 06 34 DD 7E 00 DD 23 : 58
3550 FE 41 DA 22 40 FE 5B D2 : A6
3558 22 40 D6 41 87 87 26 00 : AD
3560 6F 44 4D 29 29 09 01 E6 : 42
3568 47 09 DD 7E 00 FE 31 D8 : B2
3570 FE 3A D0 DD 23 D6 30 87 : 95
3578 85 30 01 24 6F C9 CD 85 : 64
-------------------------------------
SUM: 39 8C C3 39 0B 52 22 87 1BBD

3580 35 E5 C3 06 34 21 00 00 : 38
3588 DD 7E 00 FE 30 D8 FE 47 : A6
3590 D0 FE 3A 38 05 FE 41 D8 : 5C
3598 D6 07 D6 30 29 29 29 29 : 87
35A0 85 30 01 24 6F DD 23 18 : 61
35A8 DF D1 E1 19 E5 C3 06 34 : 8C
35B0 D1 E1 B7 ED 52 E5 C3 06 : 56
35B8 34 D1 E1 CD 33 36 E5 C3 : C4
35C0 06 34 D1 E1 CD 46 36 E5 : 1A
35C8 C3 06 34 D1 E1 CD 46 36 : F8
35D0 D5 C3 06 34 D1 E1 CD 46 : 97
35D8 36 E5 D5 C3 06 34 E1 D1 : 9F
35E0 B7 ED 52 21 00 00 20 01 : 38
35E8 23 E5 C3 06 34 D1 E1 B7 : 6E
35F0 ED 52 21 01 00 38 01 2B : C5
35F8 E5 C3 06 34 E1 D1 18 EF : 9B
-------------------------------------
SUM: A1 E4 69 68 05 DD 7D 61 697D
```

```
3600 E1 D1 B7 ED 52 21 01 00 : CA
3608 20 01 2B E5 C3 06 34 D1 : FF
3610 E1 7D A3 6F 7C A2 67 E5 : DA
3618 C3 06 34 D1 E1 7D B3 6F : 4E
3620 7C B2 67 E5 C3 06 34 D1 : 48
3628 E1 7D AB 6F 7C AA 67 E5 : EA
3630 C3 06 34 4D 44 3E 10 21 : FD
3638 00 00 29 CB 23 CB 12 30 : 24
3640 01 09 3D 20 F5 C9 4B 42 : B2
3648 5D 54 3E 10 21 00 00 CB : EB
3650 23 CB 12 ED 6A E5 B7 ED : E0
3658 42 E1 38 03 ED 42 13 3D : DD
3660 20 ED EB C9 E1 C3 06 34 : 9F
3668 D1 E1 D5 E5 C3 06 34 E1 : 4A
3670 E5 E5 C3 06 34 E1 D1 C1 : 3A
3678 D5 E5 C5 C3 06 34 CD CA : 13
-------------------------------------
SUM: 33 2B 35 15 63 CD F9 03 0C6C

3680 1F 6F 26 00 E5 C3 06 34 : 96
3688 CD D0 1F 6F 26 00 E5 C3 : F9
3690 06 34 CD 21 20 6F 26 00 : DD
3698 E5 C3 06 34 2A D3 46 54 : 79
36A0 5D 19 19 7D 84 85 6F E5 : EB
36A8 11 54 00 19 22 D3 46 E5 : 9E
36B0 C3 06 34 D1 E1 63 CD 1B : FA
36B8 20 6F 26 00 E5 C3 06 34 : 97
36C0 E1 7D CD C1 1F C3 06 34 : 08
36C8 E1 CD BE 1F C3 06 34 E1 : 69
36D0 CD 0E 40 C3 06 34 E1 7D : 76
36D8 CD F4 1F C3 06 34 E1 7E : 3C
36E0 FE 22 CA 06 34 FE 0D CA : F9
36E8 06 34 CD F4 1F 23 18 EF : 44
36F0 E1 7E FE 22 CA 06 34 FE : 81
36F8 0D CA 06 34 0E 1C FE 52 : 8B
-------------------------------------
SUM: 76 02 10 E1 DA D9 48 07 B906

3700 28 1B 0C FE 4C 28 16 0C : E3
3708 FE 55 28 11 0C FE 44 28 : 02
3710 0C 0E 0C FE 43 28 06 0C : A1
3718 FE 2F C2 22 40 79 CD F4 : 8B
3720 1F 23 18 CD 3E 0D CD F4 : 33
3728 1F C3 06 34 CD 5D 3F CD : 52
3730 4B 35 5E 23 56 13 72 2B : 07
3738 73 C3 06 34 CD 5D 3F CD : A6
3740 4B 35 5E 23 56 1B 72 2B : 0F
3748 73 C3 06 34 E1 7D CD 30 : CB
3750 20 C3 06 34 E1 45 78 B7 : 72
3758 CA 06 34 CD C4 1F 10 FB : BF
3760 C3 06 34 D1 E1 63 CD 1E : FD
3768 20 C3 06 34 ED 73 E4 46 : A7
3770 ED 7B E2 46 DD E5 ED 73 : B2
3778 E2 46 ED 7B E4 46 C3 06 : 83
-------------------------------------
SUM: 86 D6 2B A5 74 9E 12 D7 4482

3780 34 D1 2A E2 46 01 00 AE : 06
3788 B7 ED 42 D2 2E 40 7A B3 : 53
3790 CA 9D 37 2A E2 46 23 23 : 36
3798 22 E2 46 18 E1 2A E2 46 : 95
37A0 5E 23 56 D5 DD E1 18 D6 : 58
37A8 CD 5D 3F CD 6B 3F 7C FE : 5A
37B0 02 30 0F 29 CD 03 39 78 : EB
37B8 B1 CA C8 37 C5 DD E1 C3 : C0
37C0 06 34 FE 08 D2 4C 40 29 : C7
37C8 DD 22 CF 46 AF CB 3C CB : 95
37D0 1D 22 CD 46 DD 2A BF 33 : 4B
37D8 DD 7E 00 B7 28 22 FE 25 : 7F
37E0 C2 F5 37 DD 23 CD 6B 3F : 65
37E8 ED 5B CD 46 B7 ED 52 20 : 71
37F0 04 19 C3 DD 34 DD 7E 00 : 4C
37F8 DD 23 FE 0D 28 DA 18 F5 : 1A
-------------------------------------
SUM: 22 39 B4 50 CD 85 B9 79 8A38

3800 DD 2A CF 46 C3 34 40 E1 : 34
3808 7D B4 28 03 C3 06 34 DD : 36
3810 7E 00 DD 23 FE 0D CA 06 : 59
3818 34 18 F4 CD 5D 3F CD 6B : E1
3820 3F ED 73 E4 46 ED 7B E2 : 13
3828 46 DD E5 ED 73 E2 46 ED : 7D
3830 7B E4 46 C3 AE 37 2A E2 : 59
3838 46 01 00 AE B7 ED 42 D2 : AD
3840 2E 40 ED 73 E4 46 ED 7B : 60
3848 E2 46 DD E1 ED 73 E2 46 : 6E
3850 ED 7B E4 46 C3 06 34 E1 : 70
3858 D1 ED 73 E4 46 ED 7B E2 : A5
3860 46 DD E5 E5 D5 ED 73 E2 : 04
3868 46 ED 7B E4 46 C3 06 34 : D5
3870 2A E2 46 01 FC AD B7 ED : A0
3878 42 D2 2E 40 ED 73 E4 46 : 0C
-------------------------------------
SUM: 18 11 5B 03 DD F5 CA 7F 9921

3880 ED 7B E2 46 D1 E1 FD E1 : 20
3888 13 B7 ED 52 38 14 19 FD : 6B
3890 E5 E5 D5 ED 73 E2 46 ED : 14
3898 7B E4 46 FD E5 DD E1 C3 : 08
38A0 06 34 ED 73 E2 46 ED 7B : 2A
38A8 E4 46 C3 06 34 11 00 00 : 38
38B0 2A E2 46 19 5E 23 56 D5 : 17
38B8 C3 06 34 11 06 00 18 F0 : 1C
38C0 E1 ED 73 E4 46 ED 7B E2 : B5
38C8 46 E5 ED 73 E2 46 ED 7B : 1B
38D0 E4 46 C3 06 34 2A E2 46 : 79
38D8 01 00 AE B7 ED 42 D2 2E : 95
38E0 40 ED 73 E4 46 ED 7B E2 : 14
38E8 46 E1 ED 73 E2 46 ED 7B : 17
38F0 E4 46 E5 C3 06 34 2A E2 : 18
38F8 46 5E 23 56 23 73 23 72 : 48
-------------------------------------
SUM: F3 E7 4D A9 75 A7 69 50 ED86
```


▶ポピュラスは非常に面白い(ハマる)。でもなんとなく自分が神でなく，地上げ屋になったような気持ちになる。これは本当。　　大仲 忠 (26) X68000ACE 三重県

```
3900 C3 06 34 D5 E5 11 EE 49 : FF
3908 19 4E 23 46 E1 D1 C9 D5 : 20
3910 E5 11 EE 49 19 71 23 70 : 4A
3918 E1 D1 C9 C1 2A DA 46 ED : 73
3920 5B DC 46 3A DE 46 ED 43 : 0B
3928 2B 39 CD 00 00 22 DA 46 : 73
3930 ED 53 DC 46 32 DE 46 C3 : 7B
3938 06 34 E1 7D 32 DE 46 C3 : B1
3940 06 34 3A DE 46 6F 26 00 : 2D
3948 E5 C3 06 34 D1 ED 53 DC : CF
3950 46 C3 06 34 ED 5B DC 46 : AD
3958 D5 C3 06 34 E1 22 DA 46 : F5
3960 C3 06 34 2A DA 46 E5 C3 : EF
3968 06 34 E1 5E 16 00 D5 C3 : 27
3970 06 34 E1 5E 23 56 D5 C3 : 8A
3978 06 34 E1 D1 73 C3 06 34 : 5C
---------------------------------
SUM: F6 F1 01 53 B6 89 37 6F CF52

3980 E1 D1 73 23 72 C3 06 34 : B7
3988 C1 ED 58 16 00 D5 C3 06 : BA
3990 34 E1 C1 ED 69 C3 06 34 : 29
3998 E1 6C 26 00 E5 C3 06 34 : 55
39A0 E1 26 00 E5 C3 06 34 E1 : CA
39A8 7D 6C 67 E5 C3 06 34 E1 : 13
39B0 7D 2F 6F 7C 2F 67 E5 C3 : D5
39B8 06 34 C1 E1 41 CB 3C CB : EF
39C0 1D 10 FA E5 C3 06 34 C1 : CA
39C8 E1 41 29 10 FD E5 C3 06 : 06
39D0 34 CD 18 20 26 00 E5 C3 : 07
39D8 06 34 CD 18 20 6C 26 00 : D1
39E0 E5 C3 06 34 E1 CD EC 39 : B5
39E8 E5 C3 06 34 7D 2F 6F 7C : 79
39F0 2F 67 23 C9 7B 2F 5F 7A : 05
39F8 2F 57 7D 2F 6F 7C 2F 67 : B3
---------------------------------
SUM: F8 96 FD DA 04 5A 49 12 A146

3A00 01 01 00 EB 09 EB 0B ED : D9
3A08 4A C9 E1 D1 1A 77 FE 0D : 61
3A10 28 0A FE 22 28 04 23 13 : B4
3A18 18 F2 36 0D C3 06 34 C1 : 0B
3A20 E1 D1 1A 77 FE 0D 28 0D : 83
3A28 FE 22 28 07 23 13 0B 78 : 08
3A30 B1 20 EF 36 0D C3 06 34 : 00
3A38 C1 FD E1 E1 5D 54 7E FE : AD
3A40 22 28 07 FE 0D 28 03 23 : AA
3A48 18 F4 2B E5 B7 ED 52 E1 : F3
3A50 28 05 0B 78 B1 20 F3 FD : 71
3A58 E5 D1 EB 18 AF D9 C1 D9 : DB
3A60 C1 E1 D1 0B 78 B1 28 0E : DD
3A68 1A FE 22 CA 22 40 FE 0D : 71
3A70 CA 22 40 13 18 ED D9 C5 : E2
3A78 D9 C1 1A 77 FE 22 28 0B : 7E
---------------------------------
SUM: A1 8A 9C 52 6D B1 47 4A 4FB8

3A80 FE 0D 28 07 23 13 0B 78 : F3
3A88 B1 20 EF 36 0D C3 06 34 : 00
3A90 D1 E1 7E FE 0D CA 0C 3A : 4B
3A98 FE 22 CA 0C 3A 23 18 F2 : 5D
3AA0 E1 01 00 00 7E FE 0D 28 : 93
3AA8 08 FE 22 28 04 23 03 18 : 92
3AB0 F3 C5 C3 06 34 C1 E1 11 : 68
3AB8 01 00 7E B9 28 0F FE 0D : 7A
3AC0 28 08 FE 22 28 04 13 23 : B2
3AC8 18 F0 11 00 00 D5 C3 06 : B7
3AD0 34 E1 D1 1A FE 0D 28 0C : 3F
3AD8 FE 22 28 08 46 90 20 13 : 59
3AE0 23 13 18 EF 01 00 00 7E : BC
3AE8 FE 0D 28 0F FE 22 28 0B : 95
3AF0 0B 18 08 01 01 00 30 03 : 60
3AF8 01 FF FF C5 C3 06 34 C1 : 82
---------------------------------
SUM: FA 26 11 36 84 52 CE CB CEC8

3B00 D1 E1 09 EB C1 ED 4A E5 : 83
3B08 D5 C3 06 34 D1 C1 E1 B7 : FC
3B10 ED 52 EB E1 ED 42 E5 D5 : F4
3B18 C3 06 34 D1 E1 D9 FD 21 : A6
3B20 00 00 21 00 00 D1 C1 D9 : 8C
3B28 06 20 CB 3C CB 1D CB 1A : FA
3B30 CB 1B D9 30 04 FD 19 ED : F6
3B38 4A CB 23 CB 12 CB 11 CB : BC
3B40 10 D9 10 E6 D9 E5 FD E5 : 7F
3B48 C3 06 34 CD 6D 3B D9 C5 : 10
3B50 D9 C5 C3 06 34 CD 6D 3B : 10
3B58 D9 E5 D9 E5 C3 06 34 CD : 46
3B60 6D 3B D9 E5 D9 E5 D9 C5 : C2
3B68 D9 C5 C3 06 34 FD E1 D1 : 4A
3B70 21 00 00 D9 D1 21 00 00 : EC
3B78 D9 C1 D9 C1 D9 3E 20 F5 : 60
---------------------------------
SUM: 36 4C 6B 2B 35 B3 14 7A A72A

3B80 CB 21 CB 10 D9 CB 11 CB : 47
3B88 10 D9 ED 6A D9 ED 6A D9 : 49
3B90 B7 ED 52 D9 ED 52 D9 38 : 1F
3B98 0E 03 78 B1 20 03 D9 03 : 39
3BA0 D9 F1 3D 20 DA FD E9 19 : 00
3BA8 D9 ED 5A D9 18 F3 C1 D1 : 96
3BB0 E1 CD BC 3B D9 D5 D9 E5 : 11
3BB8 D5 C3 06 34 F5 C5 D9 C1 : 26
3BC0 21 00 00 11 00 00 D9 3E : 49
3BC8 20 EB 29 EB ED 6A D9 EB : 3A
3BD0 ED 6A EB ED 6A D5 E5 CD : 20
3BD8 EA 3B E1 D1 38 03 CD EA : C9
3BE0 3B D9 38 01 1C 3D 20 E1 : A7
3BE8 F1 C9 EB B7 ED 42 EB D0 : 46
3BF0 67 7D D6 01 6F 7C 26 00 : CC
3BF8 C9 C1 D1 21 00 00 D9 11 : 66
---------------------------------
SUM: 7C C8 9A 00 86 D4 F7 11 8C00

3C00 00 00 21 00 00 D9 3E 10 : 48
3C08 CB 38 CB 19 30 05 19 D9 : 0E
3C10 ED 5A D9 CB 23 CB 12 D9 : C4
3C18 CB 13 CB 12 D9 3D 20 E8 : D9
3C20 D9 E5 D9 E5 C3 06 34 C1 : 3A
3C28 D1 21 00 00 79 B7 28 0C : 56
3C30 CB 3F 30 01 19 CB 23 CB : 0D
3C38 12 C3 2D 3C E5 C3 06 34 : 20
3C40 C1 D1 E1 B7 ED 42 4D 44 : EA
3C48 E1 ED 52 38 0A 7C B5 B0 : 43
3C50 B1 20 09 E5 C3 06 34 21 : DD
3C58 FF FF 18 F7 21 01 00 18 : 47
3C60 F2 E1 11 00 00 CB 7C 28 : 53
3C68 01 1B D5 E5 C3 06 34 E1 : B4
3C70 11 00 00 7E 23 FE 0D 28 : E5
3C78 08 FE 22 28 04 53 5F 18 : 1E
---------------------------------
SUM: 68 84 22 6E 2B 18 60 EC 7A6C

3C80 F2 D5 C3 06 34 D1 E1 01 : 77
3C88 01 00 CB 7A 20 0E CB 7C : BB
3C90 20 06 B7 ED 52 38 01 0B : 60
3C98 C5 C3 06 34 CB 7C 28 F7 : 28
3CA0 18 F0 E1 11 01 00 7C B5 : 2C
3CA8 28 01 1B D5 C3 06 34 E1 : F7
3CB0 23 E5 C3 06 34 E1 2B E5 : F6
3CB8 C3 06 34 D1 21 00 00 CD : BC
3CC0 2B 3F 11 4F 3F CD E8 1F : DD
3CC8 C3 06 34 D1 E1 18 F0 E1 : 98
3CD0 11 00 00 CB 7C 28 08 3E : C6
3CD8 2D CD F4 1F CD EC 39 EB : EA
3CE0 18 DD D1 E1 CB 7C 28 D7 : ED
3CE8 3E 2D CD F4 1F CD F4 39 : 45
3CF0 C3 BF 3C E1 D1 E5 21 00 : 76
3CF8 00 CD 2B 3F E1 11 4F 3F : B7
---------------------------------
SUM: 43 22 7C 5D 8F B2 55 3F 4CB4

3D00 C3 0C 3A C1 D1 E1 C5 18 : 59
3D08 F0 D1 E1 CD BE 1F EB CD : 04
3D10 BE 1F C3 06 34 E1 DD E5 : 7D
3D18 E5 DD E1 DD 7E 00 FE 2D : 29
3D20 28 09 CD 6B 3F DD E1 E5 : 4B
3D28 C3 06 34 DD 23 CD 6B 3F : 74
3D30 CD EC 39 18 F0 E1 DD E5 : 9D
3D38 E5 DD E1 CD 89 3F DD E1 : F6
3D40 E5 D5 C3 06 34 E1 DD E5 : 5A
3D48 E5 DD E1 CD 85 35 DD E1 : E8
3D50 E5 C3 06 34 CD 18 20 26 : 0D
3D58 00 ED 5B 76 1F CD D3 1F : 9C
3D60 1A FE 1B 28 0C 19 EB E1 : 4C
3D68 1A B7 28 06 77 13 23 18 : C4
3D70 F7 E1 36 0D C3 06 34 C1 : D9
3D78 D1 E1 ED B0 C3 06 34 C1 : 0D
---------------------------------
SUM: 9E 8A 45 06 CA DE B4 67 AA5E

3D80 D1 E1 ED B8 C3 06 34 D1 : 25
3D88 C1 E1 0B 73 54 5D 13 ED : D1
3D90 B0 C3 06 34 D1 E1 E5 D5 : 19
3D98 E5 D5 C3 06 34 E1 E1 C3 : 3C
3DA0 06 34 D1 E1 D9 D1 E1 D9 : 50
3DA8 E5 D5 D9 E5 D5 C3 06 34 : 4A
3DB0 CD CD 1F C2 06 34 C3 07 : 7F
3DB8 41 DD E5 CD 00 AF DD 21 : 7D
3DC0 CA 3D CD 04 B0 DD E1 C3 : 09
3DC8 06 34 06 00 00 00 00 7F : BF
3DD0 02 C7 00 0A 00 01 02 03 : D9
3DD8 04 05 06 07 07 02 02 09 : 2A
3DE0 07 02 01 09 07 02 00 09 : 25
3DE8 0F D1 21 FA 3D 73 23 D1 : 9F
3DF0 73 DD E5 DD 21 F9 3D 18 : 81
3DF8 C9 07 02 00 0F D1 7B FE : 2B
---------------------------------
SUM: 48 01 51 AF FB BB 54 C9 2562

3E00 03 30 10 32 1D 3E DD E5 : 92
3E08 DD 21 1B 3E CD 04 B0 DD : B5
3E10 E1 18 DE DD E5 DD 21 DC : 73
3E18 3D 18 F1 07 02 00 09 0F : 67
3E20 21 F4 3E 36 0F 06 08 2B : D1
3E28 D1 73 10 FB 2B 36 0A DD : 97
3E30 E5 DD 21 EB 3E CD 04 B0 : 8D
3E38 DD E1 C3 06 34 58 CB 23 : 01
3E40 16 00 19 23 36 0F D1 2B : 93
3E48 72 2B 73 10 F9 18 E0 21 : 32
3E50 EB 3E 36 06 06 04 18 E5 : 6C
3E58 21 EB 3E 36 00 23 36 02 : DB
3E60 06 04 18 D9 21 EB 3E 36 : 7B
3E68 01 06 06 18 D0 21 EB 3E : 3F
3E70 36 02 06 04 18 C7 21 88 : CA
3E78 3E D1 72 2B 73 D1 2B 72 : 8D
---------------------------------
SUM: C1 D7 C2 05 2E 72 0C 29 9202

3E80 2B 73 C3 06 34 FF FF FF : 98
3E88 FF 21 EB 3E 36 04 06 04 : 8D
3E90 23 C5 11 85 3E 01 04 00 : C1
3E98 EB ED B0 EB C1 2B 18 9D : 14
3EA0 21 EB 3E 36 03 06 06 18 : A7
3EA8 E7 21 EB 3E 36 05 06 03 : 75
3EB0 18 DE 21 EB 3E 36 00 23 : 99
3EB8 36 01 06 02 C3 3D 3E DD : 5A
3EC0 E3 C3 C2 3D 21 EB 3E 36 : 25
3EC8 08 C1 D1 23 73 23 72 23 : E8
3ED0 71 23 70 23 36 0F DD E5 : 2E
3ED8 DD 21 EB 3E CD 04 B0 DD : 85
3EE0 E1 3A 02 C2 6F 26 00 E5 : 59
3EE8 C3 06 34 00 00 00 00 00 : FD
3EF0 00 00 00 00 00 00 00 00 : 00
3EF8 00 00 00 00 00 00 00 00 : 00
---------------------------------
SUM: 6B 39 E3 98 A9 F4 A8 BB 03F6

3F00 00 00 00 00 00 00 00 00 : 00
3F08 00 00 00 CD D9 1F C3 06 : 8E
3F10 34 CD D6 1F C3 06 34 E1 : D4
3F18 CD 94 1F 6F 26 00 E5 C3 : BD
3F20 06 34 C1 E1 79 CD 9A 1F : DB
3F28 C3 06 34 01 00 00 C5 01 : C4
3F30 0A 00 CD BC 3B D9 3E 30 : 15
3F38 83 D9 C1 03 F5 7C B5 B2 : F8
3F40 B3 20 EB 41 21 4F 3F F1 : 9F
3F48 77 23 10 FB 36 0D C9 00 : B1
3F50 00 00 00 00 00 00 00 00 : 00
3F58 00 00 00 DD 23 DD 7E 00 : 5B
3F60 FE 20 CA 5B 3F FE 09 CA : 53
3F68 5B 3F C9 21 00 00 DD 7E : DF
3F70 00 FE 30 D8 FE 3A D0 DD : EB
3F78 23 29 54 5D 29 29 19 D6 : 3E
---------------------------------
SUM: FD 3D 8A C6 4B E1 83 98 8E53

3F80 30 85 30 01 24 6F C3 6E : AA
3F88 3F 21 00 00 11 00 00 DD : 4E
3F90 7E 00 FE 30 D8 FE 3A D0 : 8C
3F98 DD 23 D6 30 EB 29 EB ED : F2
3FA0 6A 44 4D C5 42 4B EB 29 : 61
3FA8 EB ED 6A EB 29 EB ED 6A : 98
3FB0 EB 09 EB C1 ED 4A 06 00 : DD
3FB8 4F EB 09 EB 0E 00 ED 4A : 73
3FC0 18 CD DD E5 DD 21 FF 3F : E3
3FC8 FD 21 09 40 3E 04 06 05 : B4
3FD0 0E 2F DD 5E 00 DD 56 01 : AC
3FD8 0C B7 ED 52 30 FA 19 B7 : FC
3FE0 C4 F1 3F FD 71 00 DD 23 : 62
3FE8 DD 23 FD 23 10 E2 DD E1 : D0
3FF0 C9 3D 08 3E 30 B9 20 04 : 59
3FF8 08 0E 20 C9 08 AF C9 10 : 8F
---------------------------------
SUM: FA 21 C3 B9 62 5C CA F9 C855

4000 27 E8 03 64 00 0A 00 01 : 81
4008 00 00 00 00 00 00 CD C2 : 8F
4010 3F FD 21 09 40 06 05 FD : AE
4018 7E 00 CD F4 1F FD 23 10 : 8E
4020 F6 C9 11 58 40 C3 AA 40 : 15
4028 11 65 40 C3 AA 40 11 74 : E8
4030 40 C3 AA 40 ED 7B CB 46 : 66
4038 CD EB 1F 2A CD 46 CD 0E : EF
4040 40 3E 2D CD F4 1F 11 87 : 23
4048 40 C3 B1 40 11 97 40 C3 : 9F
4050 AA 40 11 A4 40 C3 AA 40 : 8C
4058 53 59 4E 54 41 58 20 45 : 4C
4060 52 52 4F 52 00 53 54 41 : 2D
4068 43 4B 20 45 4D 50 54 59 : 3D
4070 20 20 20 00 52 45 54 55 : A0
4078 52 4E 20 53 54 41 43 4B : 36
---------------------------------
SUM: 7C 66 F7 D5 7C CB A2 E1 055B

4080 20 45 4D 50 54 59 00 55 : 04
4088 4E 44 45 46 49 4E 45 44 : 3D
4090 20 4C 41 42 45 4C 00 4F : CF
4098 55 54 20 4F 46 20 4C 41 : 0B
40A0 42 45 4C 00 42 52 45 41 : ED
40A8 4B 00 ED 7B CB 46 CD EB : 7C
40B0 1F CD E5 1F CD EB 1F ED : B4
40B8 5B BF 33 DD E5 E1 AF ED : 8C
40C0 52 DA 07 41 21 01 00 22 : B8
40C8 CD 46 DD E5 E1 2B ED 53 : 21
40D0 D1 46 AF ED 52 CA F5 40 : 04
40D8 19 1A FE 0D 28 03 13 18 : 94
40E0 F8 AF ED 52 28 0F 38 0D : 62
40E8 19 ED 4B CD 46 03 ED 43 : 97
40F0 CD 46 13 18 D9 2A CD 46 : 54
40F8 CD 0E 40 3E 3A CD F4 1F : 73
---------------------------------
SUM: 9E 6A 60 33 E4 79 4C B1 62B7

4100 ED 5B D1 46 CD E8 1F ED : 20
4108 7B CB 46 CD EB 1F C3 31 : 57
4110 30 CD B2 1F DA B5 33 E5 : 75
4118 FD E1 DD 2A BF 33 DD E5 : 99
4120 D1 CD E8 1F CD EE 1F DD : 5C
4128 7E 00 B7 CA A5 42 CD 5D : 10
4130 3F DD 7E 00 FD 77 00 DD : EB
4138 23 B7 CA A5 42 FE 27 CA : 7A
4140 A7 41 FE 24 CA C0 41 FE : D3
4148 22 CA EF 41 FE 23 CA 14 : 1B
4150 42 FE 25 CA 5F 42 FE 2E : FC
4158 CA 2F 42 FE 3B CA C5 41 : 44
4160 FE 2D CA 8C 41 FE 0D CA : 97
4168 D1 41 FE 30 DA 63 42 FE : BD
4170 3A D2 63 42 DD 2B CD 6B : F1
4178 3F 0E 8F FD 71 00 FD 75 : BC
---------------------------------
SUM: 63 BB 9B 12 CD 0F EC F2 A50E

4180 01 FD 74 02 FD 23 FD 23 : B4
4188 FD 23 18 A2 47 DD 7E 00 : 7C
4190 FE 20 78 CA 63 42 DD 7E : 60
4198 00 FE 09 78 CA 63 42 CD : BB
41A0 6B 3F CD EC 39 18 D2 CD : 53
41A8 89 3F FD 36 00 94 FD 75 : 01
41B0 01 FD 74 02 FD 73 03 FD : E4
41B8 72 04 FD 23 FD 23 18 C4 : 92
41C0 CD 85 35 18 B4 DD 7E 00 : AE
41C8 FE 0D CA 2E 41 DD 23 18 : 5C
41D0 F4 FD 36 00 FF FD 23 DD : 23
41D8 7E 00 B7 CA 2E 41 DD E5 : 30
41E0 D1 CD E8 1F CD EE 1F CD : 4C
41E8 C7 1F D4 42 C3 2E 41 FD : 2B
41F0 36 00 8A FD 23 DD 7E 00 : 3B
41F8 FD 77 00 FE 22 28 0A FE : C4
---------------------------------
SUM: 6B AF 7A 99 9B 00 0D 13 DF2D

4200 0D 28 06 DD 23 FD 23 18 : 73
4208 EC FD 36 00 22 FD 23 DD : 3E
4210 23 C3 2E 41 0E 8E DD 7E : 4C
4218 00 FE 23 20 04 DD 23 0E : 53
4220 93 C5 CD 4B 35 B7 11 E6 : 53
4228 47 ED 52 C1 C3 7B 41 DD : A3
```

THE SENTINEL

▶バンダイのプラモデルのパトレイバーシリーズはなかなかよい。しかし，モールドが浅いようだ。しかも塗装をするとなるとメーンカメラのカバー(?)とゴム部分のマスキングが難しい。などといいつつ1体も完成していないのであった。

栢　一夫(16) X68000EXPERT-HD　三重県

```
4230 7E 00 FE 2E 28 04 0E 8D : 71
4238 18 E7 DD 23 0E 92 18 E1 : 98
4240 CD 5D 3F 0E 90 18 DA CD : C6
4248 5D 3F 0E 91 18 D3 0E 8B : BF
4250 C5 CD 5D 3F CD 6B 3F C1 : 66
4258 C3 7B 41 0E 8C 18 F1 0E : 30
4260 88 18 ED 47 DD 7E 00 FE : 2D
4268 20 28 0D FE 09 28 09 FE : 8B
4270 0D 28 05 DD 23 80 18 EB : BD
4278 78 FE 39 28 D1 FE 80 28 : 4E
-----------------------------------
SUM: 6B C9 AA D1 60 BF 77 E8 39C6

4280 DA FE DA 28 BB FE CC 28 : 87
4288 BE FE 8F 20 04 3E 89 18 : 4E
4290 0C 21 19 44 85 30 01 24 : 64
4298 6F 7E B7 28 26 FD 77 00 : 66
42A0 FD 23 C3 2E 41 CD EB 1F : 29
42A8 CD E2 1F 4F 42 4A 45 43 : 31
42B0 54 20 45 4E 44 3A 00 FD : 82
42B8 E5 E1 CD BE 1F CD EB 1F : 47
42C0 C3 31 30 CD EB 1F CD E2 : AA
42C8 1F 53 59 4E 54 41 58 3F : 45
42D0 00 C3 31 30 CD EB 1F CD : C8
42D8 E2 1F 42 52 45 41 4B 00 : 66
42E0 C3 31 30 DD 7E 00 DD 23 : 7F
42E8 B7 CA 07 41 FE FF 28 F3 : E1
42F0 6F 26 00 29 11 19 45 19 : 46
42F8 5E 23 56 EB E9 DD 6E 00 : F6
-----------------------------------
SUM: 21 4B B6 0C 17 08 2F FF E944

4300 DD 66 01 E5 DD 23 DD 23 : 29
4308 18 D9 DD 5E 00 DD 56 01 : 60
4310 D5 DD 23 DD 23 18 E6 DD : B0
4318 6E 00 DD 66 01 11 E6 47 : F0
4320 19 C9 CD 17 43 5E 23 56 : E0
4328 D5 18 D9 CD 17 43 4E 23 : 5E
4330 46 23 C5 18 F0 CD 17 43 : 5D
4338 D1 73 23 72 18 C6 CD 17 : 9B
4340 43 D1 C1 71 23 70 23 18 : 14
4348 F0 CD 17 43 DD 23 DD 23 : 17
4350 C3 32 37 CD 17 43 DD 23 : 53
4358 DD 23 C3 42 37 DD 6E 00 : 87
4360 DD 66 01 C3 80 43 DD 6E : 15
4368 00 DD 66 01 DD 23 DD 23 : 44
4370 C3 F6 43 DD 6E 00 DD 66 : 8A
4378 01 DD 23 DD 23 C3 DD 34 : D5
-----------------------------------
SUM: B1 9C 0B 35 9F 39 13 A4 2082

4380 29 CD 03 39 78 B1 CA 8F : B4
4388 43 C5 DD E1 C3 E3 42 DD : 8B
4390 22 CF 46 AF CB 3C CB 1D : D5
4398 22 CD 46 DD 2A E0 46 DD : 3F
43A0 7E 00 FE 88 C2 C0 43 DD : A6
43A8 23 DD 6E 00 DD 66 01 DD : 8F
43B0 23 DD 23 ED 5B CD 46 B7 : 35
43B8 ED 52 20 04 19 C3 DD 34 : 50
43C0 CD C5 43 18 DA DD 7E 00 : 22
43C8 DD 23 FE FF C8 FE 8A 30 : 7D
43D0 02 18 F2 FE 8A 28 14 FE : CE
43D8 92 30 06 DD 23 DD 23 18 : E0
43E0 E4 DD 23 DD 23 DD 23 DD : C1
43E8 23 18 DA DD 7E 00 DD 23 : 70
43F0 FE 22 28 D1 18 F5 ED 73 : 86
43F8 E4 46 ED 7B E2 46 DD E5 : 7C
-----------------------------------
SUM: 88 C7 66 17 2D 5E 8D A9 6082

4400 ED 73 E2 46 ED 7B E4 46 : 1A
4408 C3 80 43 E1 7D B4 28 03 : C3
4410 C3 E3 42 CD C5 43 C3 E3 : 63
4418 42 00 00 00 00 00 00 00 : 42
4420 81 00 00 00 59 7C 00 00 : 56
4428 06 00 00 7B 00 57 58 00 : 30
4430 16 00 17 53 00 27 00 00 : A7
4438 1F 36 29 00 00 2B 00 00 : A9
4440 5C 2D 67 03 01 4A 02 68 : A8
4448 04 00 56 64 00 60 0E 15 : 41
4450 00 1B 21 28 10 08 2A 09 : AF
4458 6F 00 2C 3C 3D 00 55 00 : 69
4460 00 71 1A 00 7E 00 00 00 : 09
4468 40 54 00 72 7D 00 00 69 : EC
4470 00 00 00 00 35 00 00 0A : 3F
4478 00 00 6E 00 00 00 82 00 : F0
-----------------------------------
SUM: 80 19 39 FF 06 49 38 25 A9F8

4480 2E 65 4D 00 00 0F 4F 4B : 89
4488 1E 00 30 13 00 00 00 48 : A9
4490 46 00 47 07 49 2F 70 00 : 7C
4498 5F 22 5E 4E 00 00 00 31 : 5E
44A0 5D 74 75 6C 00 26 18 00 : F0
44A8 24 44 00 00 34 00 76 00 : 12
44B0 32 79 00 00 00 00 38 00 : E3
44B8 00 00 0C 41 00 00 00 78 : C5
44C0 00 00 00 00 00 00 00 00 : 00
44C8 00 00 00 6B 5A 3E 5B 63 : C1
44D0 00 20 00 00 66 00 00 50 : D6
44D8 51 00 25 00 00 00 00 00 : 76
44E0 00 00 12 00 00 1D 00 00 : 2F
44E8 00 6A 42 7A 0B 00 00 00 : 31
44F0 73 43 45 1C 00 00 19 61 : 91
44F8 00 05 00 62 4C 14 3F 00 : 06
-----------------------------------
SUM: 68 8A 61 78 94 D3 38 50 4621

4500 6D 52 11 00 23 00 3B 00 : 2E
4508 80 00 39 37 3A 00 77 00 : A1
4510 00 33 0D 00 00 00 7F 83 : 42
4518 00 22 40 A9 35 B0 35 B9 : DE
4520 35 C2 35 CB 35 D4 35 DE : 13
4528 35 ED 35 FC 35 00 36 0F : CD
4530 36 1B 36 27 36 64 36 68 : E6
4538 36 6F 36 7E 36 88 36 92 : DF
4540 36 9C 36 B3 36 C0 36 C8 : AF
4548 36 CF 36 D6 36 DE 36 F0 : 4B
4550 36 2C 37 3C 37 4C 37 54 : E3
4558 37 63 37 A8 37 1B 38 36 : 39
4560 38 07 38 6C 37 81 37 1B : ED
4568 39 3A 39 42 39 4C 39 54 : 00
4570 39 5C 39 63 39 6A 39 72 : 7F
4578 39 7A 39 80 39 88 39 91 : F7
-----------------------------------
SUM: 7F F1 2A 4A 24 34 FA D7 8FD1

4580 39 57 38 70 38 98 39 A0 : E1
4588 39 A7 39 AF 39 BA 39 C7 : BB
4590 39 D1 39 DA 39 E4 39 0A : 7D
4598 3A 1F 3A 38 3A 90 3A A0 : 6F
45A0 3A B5 3A D1 3A FF 3A 0C : 79
45A8 3B 1B 3B 4B 3B 55 3B F9 : A0
45B0 3B 61 3C 6F 3C 85 3C A2 : E6
45B8 3C BB 3C CB 3C CF 3C E2 : 27
45C0 3C F3 3C 03 3D 09 3D 15 : 06
45C8 3D 35 3D 54 3D 77 3D 7F : 73
45D0 3D 87 3D 94 3D 9D 3D A2 : 4E
45D8 3D B9 3D E9 3D FD 3D 20 : B3
45E0 3E 4F 3E 58 3E 64 3E 6D : 70
45E8 3E 76 3E A0 3E 89 3E A9 : 40
45F0 3E C4 3E B2 3E BF 3E 0B : 38
45F8 3F 11 3F 17 3F 22 3F 07 : 4D
-----------------------------------
SUM: BD DC BD 1C BE 56 BF 18 64AA

4600 41 AD 38 BB 38 24 37 75 : E9
4608 36 C0 38 D5 38 F6 38 40 : A9
4610 3C 5F 3B AE 3B 27 3C AF : D1
4618 3C B5 3C 45 3D B0 3D 5D : F9
4620 3A 00 00 00 00 00 00 00 : 3A
4628 00 73 43 0B 44 01 35 5D : 98
4630 43 66 43 35 43 22 43 FD : C6
4638 42 49 43 53 43 3E 43 2B : 10
4640 43 0A 43 00 2B 2D 2A 2F : 41
4648 4D 2F 3D 3C 3E 21 41 4F : E4
4650 58 44 53 43 4B 47 46 52 : 5C
4658 53 48 48 50 43 50 43 49 : 52
4660 44 57 42 4C 47 47 52 49 : 52
4668 52 55 43 50 47 50 47 50 : 68
4670 47 50 50 50 50 49 4F 44 : 63
4678 4C 48 4C 45 4E 52 52 43 : 5A
-----------------------------------
SUM: 12 AC EC 16 D5 69 D1 7F 005E

4680 43 4E 53 4C 52 53 53 49 : 71
4688 53 4C 4C 4C 4C 4C 44 43 : 56
4690 41 46 3D 50 50 50 50 53 : 57
4698 53 48 56 56 49 54 54 46 : 7E
46A0 43 44 53 49 43 43 50 57 : 50
46A8 4C 53 42 54 54 42 43 50 : 5E
46B0 44 4D 50 50 50 50 45 49 : 5F
46B8 4A 43 52 54 46 4C 43 4C : 54
46C0 44 2A 49 44 56 42 4D 00 : E0
46C8 00 00 00 00 00 00 00 00 : 00
46D0 00 00 00 00 00 00 00 00 : 00
46D8 00 00 00 00 00 00 00 00 : 00
46E0 00 80 00 AE 00 00 00 00 : 2E
46E8 00 00 00 00 00 00 00 00 : 00
46F0 00 00 00 00 00 00 00 00 : 00
46F8 00 00 00 00 00 00 00 00 : 00
-----------------------------------
SUM: 8B F9 B2 71 BA A6 A3 61 BEC4
```

4DFF$_H$まで0で埋める

リスト2 STACKソースリスト

```
3000                1         ORG   $3000
3000                2      OFFSET   $B000
3000                3 ;
3000                4 #PRINT  EQU   $1FF4
3000                5 #MPRINT EQU   $1FE2
3000                6 #MSG    EQU   $1FE8
3000                7 #NL     EQU   $1FEB
3000                8 #LTNL   EQU   $1FEE
3000                9 #GETL   EQU   $1FD3
3000               10 #PRTHL  EQU   $1FBE
3000               11 #HLHEX  EQU   $1FB2
3000               12 #KBFAD  EQU   $1F76
3000               13 #PAUSE  EQU   $1FC7
3000               14 #BELL   EQU   $1FC4
3000               15 #HOT    EQU   $1FFA
3000               16 #FILE   EQU   $1FA3
3000               17 #WOPEN  EQU   $1FAF
3000               18 #WRD    EQU   $1FAC
3000               19 #ROPEN  EQU   $2009
3000               20 #RDD    EQU   $1FA6
3000               21 #FPRINT EQU   $1F9D
3000               22 #DTADR  EQU   $1F70
3000               23 #SIZE   EQU   $1F72
3000               24 #EXADR  EQU   $1F6E
3000               25 #LPSW   EQU   $1F7C
3000               26 #MSX    EQU   $1FE5
3000               27 #PRTHX  EQU   $1FC1
3000               28 #BREAK  EQU   $1FCD
3000               29 #GETKY  EQU   $1FD0
3000               30 #INKEY  EQU   $1FCA
3000               31 #FLGET  EQU   $2021
3000               32 #2HEX   EQU   $1FB5
3000               33 #CSR    EQU   $2018
3000               34 #POKE   EQU   $1F9A
3000               35 #PEEK   EQU   $1F94
3000               36 #WIDCH  EQU   $2030
3000               37 #LOC    EQU   $201E
3000               38 #WIDTH  EQU   $1F5C
3000               39 #SCRN   EQU   $201B
3000               40 #DIR    EQU   $2006
3000               41 #ERROR  EQU   $2033
3000               42 @RET_SP EQU   $AE00
3000               43 @VAR_SP EQU   $AA00
3000               44 ;
3000               45 ; STACK Editor ver 1.0
3000               46 ;
3000 C3 06 30      47         JP    COLD
3003 C3 31 30      48         JP    HOT
3006               49 COLD
3006 CD E2 1F      50         CALL  #MPRINT
3009 0C            51         DB    $0C
300A 2A 20 53      52         DM    '* Stack Interpreter *"
300D 74 61 63
3010 6B 20 49
3013 6E 74 65
3016 72 70 72
3019 65 74 65
301C 72 20 2A
301F 00            53         DB    0
3020               54 COLD1
3020 2A C3 33      55         LD    HL,(SADDR)
3023 22 BF 33      56         LD    (TEXTS),HL
3026 22 C1 33      57         LD    (TEXTE),HL
3029 36 00         58         LD     (HL),0
302B 21 00 00      59         LD    HL,0
302E 22 C5 33      60         LD    (POINTE),HL
3031               61 HOT
3031               62 E_MAIN
3031 ED 73 C8      63         LD    (WRK),SP
3034 33
3035 CD EB 1F      64         CALL  #NL
3038 3E 5D         65         LD    A,']'
303A CD F4 1F      66         CALL  #PRINT
303D ED 5B 76      67         LD    DE,(#KBFAD)
3040 1F
3041 CD D3 1F      68         CALL  #GETL
3044 1A            69         LD    A,(DE)
3045 13            70         INC   DE
3046 FE 5D         71         CP    ']'
3048 C2 9E 31      72         JP    NZ,SCEDIT
304B               73 ;
304B 1A            74         LD    A,(DE)
304C 13            75         INC   DE
304D FE 49         76         CP    'I'
304F CA E3 31      77         JP    Z,INPUT
3052 FE 54         78         CP    'T'
3054 CA EA 31      79         JP    Z,LIST
3057 FE 26         80         CP    '&'
3059 28 C5         81         JR    Z,COLD1
305B FE 44         82         CP    'D'
305D CA 1C 32      83         JP    Z,DELETE
3060 FE 42         84         CP    'B'
3062 CA 56 32      85         JP    Z,INSERT
3065 FE 53         86         CP    'S'
3067 CA 77 32      87         JP    Z,SAVE
306A FE 52         88         CP    'R'
306C 28 4E         89         JR    Z,RECOVER
306E FE 4C         90         CP    'L'
3070 CA B9 32      91         JP    Z,LOAD
3073 FE 58         92         CP    'X'
3075 28 52         93         JR    Z,CHANGE
3077 FE 4D         94         CP    'M'
3079 28 5F         95         JR    Z,MEMORY
307B FE 50         96         CP    'P'
307D 28 6E         97         JR    Z,PRT
307F FE 21         98         CP    '!'
3081 CA FA 1F      99         JP    Z,#HOT
3084 FE 47        100         CP    'G'
3086 CA E3 33     101         JP    Z,RUN
3089 FE 57        102         CP    'W'
308B CA 12 31     103         JP    Z,WIDTH
308E FE 46        104         CP    'F'
3090 CA 21 31     105         JP    Z,FIND
3093 FE 5A        106         CP    'Z'
3095 CA 8B 31     107         JP    Z,DIR
3098 FE 43        108         CP    'C'
309A CA 11 41     109         JP    Z,COMP
309D FE 4A        110         CP    'J'
309F CA D3 33     111         JP    Z,CRUN
30A2 FE 20        112         CP    ' '
30A4 20 8B        113         JR    NZ,E_MAIN
30A6 21 E5 46     114         LD    HL,DIRECT_B-1
30A9              115 ;
30A9              116 ; DIRECT MODE
30A9              117 ;
30A9              118 @DIRECT
30A9 1A           119         LD    A,(DE)
30AA 13           120         INC   DE
30AB 23           121         INC   HL
30AC 77           122         LD    (HL),A
30AD B7           123         OR    A
30AE 20 F9        124         JR    NZ,@DIRECT
30B0 36 0D        125         LD    (HL),$0D
30B2 23           126         INC   HL
30B3 36 00        127         LD    (HL),0
30B5 DD 21 E6     128         LD    IX,DIRECT_B
30B8 46
30B9 C3 E7 33     129         JP    DIRECT
30BC              130 ; RECOVER
30BC              131 ;
30BC              132 RECOVER
30BC ED 5B BF     133         LD    DE,(TEXTS)
30BF 33
30C0 3E 20        134         LD    A,' '
30C2 12           135         LD    (DE),A
30C3 CD 5F 33     136         CALL  #SEARCH
30C6              137 RE1
30C6 C3 31 30     138         JP    E_MAIN
30C9              139 ;
30C9              140 ; CHANGE TEXT TOP
30C9              141 ;
30C9              142 CHANGE
30C9 CD B2 1F     143         CALL  #HLHEX
30CC DA B5 33     144         JP    C,ERR
30CF 22 C3 33     145         LD    (SADDR),HL
30D2 22 BF 33     146         LD    (TEXTS),HL
30D5 CD 5F 33     147         CALL  #SEARCH
30D8 18 EC        148         JR    RE1
30DA              149 ;
30DA              150 ; PRINT TEXT ADDRESS
30DA              151 ;
30DA              152 MEMORY
30DA 2A BF 33     153         LD    HL,(TEXTS)
30DD CD BE 1F     154         CALL  #PRTHL
30E0 3E 2D        155         LD    A,'-'
30E2 CD F4 1F     156         CALL  #PRINT
30E5 2A C1 33     157         LD    HL,(TEXTE)
30E8 CD BE 1F     158         CALL  #PRTHL
30EB 18 D9        159         JR    RE1
30ED              160 ;
30ED              161 ; PRINTER ON/OFF
30ED              162 ;
30ED              163 PRT
30ED CD EB 1F     164         CALL  #NL
30F0 CD E2 1F     165         CALL  #MPRINT
30F3 50 52 49     166         DM    "PRINTER "
30F6 4E 54 45
30F9 52 20
30FB 00           167         DB    0
30FC 3A C7 33     168         LD    A,(LPFLAG)
30FF EE 01        169         XOR   1
3101 32 C7 33     170         LD    (LPFLAG),A
3104 11 CC 33     171         LD    DE,@PRTON
3107 B7           172         OR    A
3108 20 03        173         JR    NZ,PRTON
310A 11 CF 33     174         LD    DE,@PRTOFF
310D              175 PRTON
310D CD E5 1F     176         CALL  #MSX
3110 18 B4        177         JR    RE1
3112              178 ;
3112              179 ; WIDTH CHNAGE
3112              180 ;
3112              181 WIDTH
3112 3A 5C 1F     182         LD     A,(#WIDTH)
3115 FE 28        183         CP     40
```


▶ゲームは買わないという約束でパソコンを買ったため毎年4月号を見て後悔しています。親に見つからないように「テトリス」をして過ごしています。
坂本　浩一(16) X1turboII　京都府

```
3117 28 02      184         JR   Z,WIDTH1
3119 3E 14      185         LD   A,20
311B            186 WIDTH1
311B 87         187         ADD  A,A
311C CD 30 20   188         CALL #WIDCH
311F 18 A5      189         JR   RE1
3121            190 ;
3121            191 ; FIND STRINGS
3121            192 ;
3121            193 FIND
3121 DD 2A BF   194         LD   IX,(TEXTS)
3124 33
3125 01 00 00   195         LD   BC,0
3128 ED 43 CD   196         LD   (LINE_WR),BC
312B 46
312C            197 FIND1
312C DD 22 D1   198         LD   (LINE_TOP),IX
312F 46
3130 ED 4B CD   199         LD   BC,(LINE_WR)
3133 46
3134 03         200         INC  BC
3135 ED 43 CD   201         LD   (LINE_WR),BC
3138 46
3139            202 FIND2
3139 CD C7 1F   203         CALL #PAUSE
313C 31 30      204         DW   E_MAIN
313E DD 7E 00   205         LD   A,(IX)
3141 DD 23      206         INC  IX
3143 FE 0D      207         CP   $0D
3145 28 E5      208         JR   Z,FIND1
3147 B7         209         OR   A
3148 CA 31 30   210         JP   Z,E_MAIN
314B 47         211         LD   B,A
314C ED 5B 76   212         LD   DE,(#KBFAD)
314F 1F
3150 13         213         INC  DE
3151 13         214         INC  DE
3152 1A         215         LD   A,(DE)
3153 B8         216         CP   B
3154 20 E3      217         JR   NZ,FIND2
3156 DD E5      218         PUSH IX
3158            219 FIND3
3158 13         220         INC  DE
3159 1A         221         LD   A,(DE)
315A B7         222         OR   A
315B 28 0C      223         JR   Z,PRTFIND
315D DD 46 00   224         LD   B,(IX)
3160 DD 23      225         INC  IX
3162 B8         226         CP   B
3163 28 F3      227         JR   Z,FIND3
3165 DD E1      228         POP  IX
3167 18 D0      229         JR   FIND2
3169            230 PRTFIND
3169 DD E1      231         POP  IX
316B 2A CD 46   232         LD   HL,(LINE_WR)
316E CD 0E 40   233         CALL @DEC2
3171 3E 3A      234         LD   A,':'
3173 CD F4 1F   235         CALL #PRINT
3176 ED 5B D1   236         LD   DE,(LINE_TOP)
3179 46
317A CD E8 1F   237         CALL #MSG
317D CD EB 1F   238         CALL #NL
3180            239 FIND4
3180 DD 7E 00   240         LD   A,(IX)
3183 DD 23      241         INC  IX
3185 FE 0D      242         CP   $0D
3187 20 F7      243         JR   NZ,FIND4
3189 18 A1      244         JR   FIND1
318B            245 ;
318B            246 ; DIR
318B            247 ;
318B            248 DIR
318B CD 06 20   249         CALL #DIR
318E C3 31 30   250         JP   E_MAIN
3191            251 ;
3191            252 ; SCREEN EDIT
3191            253 ;
3191            254 ETABSUB
3191 E5         255         PUSH HL
3192 CD 18 20   256         CALL #CSR
3195 3E 06      257         LD   A,06
3197 85         258         ADD  A,L
3198 6F         259         LD   L,A
3199 CD 1E 20   260         CALL #LOC
319C E1         261         POP  HL
319D C9         262         RET
319E            263 SCEDIT
319E 1B         264         DEC  DE
319F            265 SCEDIT1
319F CD 7C 33   266         CALL TEXTSKIP
31A2 D5         267         PUSH DE
31A3 01 01 00   268         LD   BC,1
31A6 CD FB 32   269         CALL #DEL
31A9 ED 5B 76   270         LD   DE,(#KBFAD)
31AC 1F
31AD 13         271         INC  DE
31AE 13         272         INC  DE
31AF 13         273         INC  DE
31B0 13         274         INC  DE
31B1 13         275         INC  DE
31B2 1A         276         LD   A,(DE)
31B3 FE 3A      277         CP   ':'
31B5 C2 B5 33   278         JP   NZ,ERR
31B8 CD 91 31   279         CALL ETABSUB
31BB 13         280         INC  DE
31BC 1A         281         LD   A,(DE)
31BD CD 4F 33   282         CALL #COUNT
31C0 D1         283         POP  DE
31C1 CD 31 33   284         CALL #INS
31C4 2A 76 1F   285         LD   HL,(#KBFAD)
31C7 23         286         INC  HL
31C8 23         287         INC  HL
31C9 23         288         INC  HL
31CA 23         289         INC  HL
31CB 23         290         INC  HL
31CC 23         291         INC  HL
31CD ED B0      292         LDIR
31CF ED 5B 76   293         LD   DE,(#KBFAD)
31D2 1F
31D3 CD D3 1F   294         CALL #GETL
31D6 1A         295         LD   A,(DE)
31D7 FE 20      296         CP   ' '
31D9 DA C6 30   297         JP   C,RE1
31DC FE 5D      298         CP   ']'
31DE CA 31 30   299         JP   Z,E_MAIN
31E1 18 BC      300         JR   SCEDIT1
31E3            301 ;
31E3            302 ;  APPEND MODE
31E3            303 ;
31E3            304 INPUT
31E3 ED 5B C1   305         LD   DE,(TEXTE)
31E6 33
31E7 C3 59 32   306         JP   INS
31EA            307 ;
31EA            308 ; LIST
31EA            309 ;
31EA            310 LIST
31EA CD 7C 33   311         CALL TEXTSKIP
31ED            312 LS1
31ED 3A C7 33   313         LD   A,(LPFLAG)
31F0 32 7C 1F   314         LD   (#LPSW),A
31F3 D5         315         PUSH DE
31F4 E5         316         PUSH HL
31F5 CD 0E 40   317         CALL @DEC2
31F8 E1         318         POP  HL
31F9 D1         319         POP  DE
31FA 3E 3A      320         LD   A,':'
31FC CD F4 1F   321         CALL #PRINT
31FF CD E8 1F   322         CALL #MSG
3202 CD EB 1F   323         CALL #NL
3205 AF         324         XOR  A
3206 32 7C 1F   325         LD   (#LPSW),A
3209            326 LS2
3209 1A         327         LD   A,(DE)
320A FE 0D      328         CP   $0D
320C 13         329         INC  DE
320D 20 FA      330         JR   NZ,LS2
320F 1A         331         LD   A,(DE)
3210 B7         332         OR   A
3211 CA 31 30   333         JP   Z,E_MAIN
3214 CD C7 1F   334         CALL #PAUSE
3217 31 30      335         DW   E_MAIN
3219 23         336         INC  HL
321A 18 D1      337         JR   LS1
321C            338 ;
321C            339 ; DELETE
321C            340 ;
321C            341 DELETE
321C CD A4 33   342         CALL #HLDECI
321F DA B5 33   343         JP   C,ERR
3222 DD 7E 00   344         LD   A,(IX)
3225 FE 2C      345         CP   ","
3227 20 21      346         JR   NZ,DELETE_1
3229 DD 23      347         INC  IX
322B E5         348         PUSH HL
322C CD A7 33   349         CALL #HLDECI_1
322F DA B5 33   350         JP   C,ERR
3232 D1         351         POP  DE
3233 ED 52      352         SBC  HL,DE
3235 DA B5 33   353         JP   C,ERR
3238 CA B5 33   354         JP   Z,ERR
323B            355 ;
323B EB         356         EX   DE,HL
323C 4B         357         LD   C,E
323D 42         358         LD   B,D
323E 03         359         INC  BC
323F C5         360         PUSH BC
3240 CD 81 33   361         CALL TEXTSKIP1
3243 C1         362         POP  BC
3244 CD FB 32   363         CALL #DEL
3247 C3 31 30   364         JP   E_MAIN
324A            365 DELETE_1
324A CD 81 33   366         CALL TEXTSKIP1
324D 01 01 00   367         LD   BC,1
3250 CD FB 32   368         CALL #DEL
3253            369 DELETE1
3253 C3 31 30   370         JP   E_MAIN
3256            371 ;
3256            372 ; INSERT MODE
3256            373 ;
3256            374 INSERT
3256 CD 7C 33   375         CALL TEXTSKIP
3259            376 INS
3259 D5         377         PUSH DE
325A ED 5B 76   378         LD   DE,(#KBFAD)
325D 1F
325E CD D3 1F   379         CALL #GETL
3261 1A         380         LD   A,(DE)
3262 FE 1B      381         CP   $1B
3264 20 03      382         JR   NZ,INS1
3266 D1         383         POP  DE
3267 18 EA      384         JR   DELETE1
3269            385 INS1
3269 CD 4F 33   386         CALL #COUNT
326C D1         387         POP  DE
326D CD 31 33   388         CALL #INS
3270 2A 76 1F   389         LD   HL,(#KBFAD)
3273 ED B0      390         LDIR
3275 18 E2      391         JR   INS
3277            392 ;
3277            393 ; TEXT SAVE
3277            394 ;
3277            395 SAVE
3277 ED 4B BF   396         LD   BC,(TEXTS)
327A 33
327B 0A         397         LD   A,(BC)
327C B7         398         OR   A
327D CA B5 33   399         JP   Z,ERR
3280 ED 43 70   400         LD   (#DTADR),BC
3283 1F
3284 2A C1 33   401         LD   HL,(TEXTE)
3287 B7         402         OR   A
3288 ED 42      403         SBC  HL,BC
328A 23         404         INC  HL
328B 22 72 1F   405         LD   (#SIZE),HL
328E 21 00 00   406         LD   HL,0
3291 22 6E 1F   407         LD   (#EXADR),HL
3294 3E 04      408         LD   A,4
3296 CD A3 1F   409         CALL #FILE
3299 CD AF 1F   410         CALL #WOPEN
329C DA B2 33   411         JP   C,ERRPRT
329F CD EB 1F   412         CALL #NL
32A2 CD E2 1F   413         CALL #MPRINT
32A5 57 52 49   414         DM   "WRITING-"
32A8 54 49 4E
32AB 47 2D
32AD 00         415         DB   0
32AE CD 9D 1F   416         CALL #FPRINT
32B1 CD AC 1F   417         CALL #WRD
32B4 DA B2 33   418         JP   C,ERRPRT
32B7            419 SAVE1
32B7 18 9A      420         JR   DELETE1
32B9            421 ;
32B9            422 ; TEXT LOAD
32B9            423 ;
32B9            424 LOAD
32B9 3E 04      425         LD   A,4
32BB CD A3 1F   426         CALL #FILE
32BE            427 LOAD1
32BE CD 09 20   428         CALL #ROPEN
32C1 DA B2 33   429         JP   C,ERRPRT
32C4 CA D8 32   430         JP   Z,LOAD2
32C7 CD EB 1F   431         CALL #NL
32CA CD E2 1F   432         CALL #MPRINT
32CD 53 4B 49   433         DM   "SKIP-"
32D0 50 2D
32D2 00         434         DB   0
32D3 CD 9D 1F   435         CALL #FPRINT
32D6 18 E6      436         JR   LOAD1
32D8            437 LOAD2
32D8 CD EB 1F   438         CALL #NL
32DB CD E2 1F   439         CALL #MPRINT
32DE 4C 4F 41   440         DM   "LOADING-"
32E1 44 49 4E
32E4 47 2D
32E6 00         441         DB   0
32E7 CD 9D 1F   442         CALL #FPRINT
32EA 2A C1 33   443         LD   HL,(TEXTE)
32ED 22 70 1F   444         LD   (#DTADR),HL
32F0 CD A6 1F   445         CALL #RDD
32F3 DA B2 33   446         JP   C,ERRPRT
32F6 CD 5F 33   447         CALL #SEARCH
32F9 18 BC      448         JR   SAVE1
32FB            449 ;
32FB            450 #DEL
32FB ED 43 CA   451         LD   (P_WRK),BC
32FE 33
32FF D5         452         PUSH DE
3300            453 DEL2
3300 1A         454         LD   A,(DE)
3301 B7         455         OR   A
3302 CA B5 33   456         JP   Z,ERR
3305 13         457         INC  DE
3306 FE 0D      458         CP   $0D
3308 28 02      459         JR   Z,DEL1'
330A 18 F4      460         JR   DEL2
330C            461 DEL1'
330C 0B         462         DEC  BC
330D 78         463         LD   A,B
330E B1         464         OR   C
330F 20 EF      465         JR   NZ,DEL2
3311            466 DEL1
3311 2A C1 33   467         LD   HL,(TEXTE)
3314 B7         468         OR   A
3315 ED 52      469         SBC  HL,DE
3317 4D         470         LD   C,L
3318 44         471         LD   B,H
3319 EB         472         EX   DE,HL
331A D1         473         POP  DE
331B 03         474         INC  BC
331C ED B0      475         LDIR
331E 1B         476         DEC  DE
331F ED 53 C1   477         LD   (TEXTE),DE
3322 33
3323 2A C5 33   478         LD   HL,(POINTE)
3326 ED 4B CA   479         LD   BC,(P_WRK)
3329 33
332A B7         480         OR   A
332B ED 42      481         SBC  HL,BC
332D 22 C5 33   482         LD   (POINTE),HL
3330 C9         483         RET
3331            484 ;
3331            485 #INS
3331 C5         486         PUSH BC
3332 D5         487         PUSH DE
3333 2A C1 33   488         LD   HL,(TEXTE)
3336 E5         489         PUSH HL
3337 09         490         ADD  HL,BC
3338 22 C1 33   491         LD   (TEXTE),HL
333B E5         492         PUSH HL
333C B7         493         OR   A
333D ED 52      494         SBC  HL,DE
333F 4D         495         LD   C,L
3340 44         496         LD   B,H
3341 D1         497         POP  DE
3342 E1         498         POP  HL
3343 ED B8      499         LDDR
3345 2A C5 33   500         LD   HL,(POINTE)
3348 23         501         INC  HL
3349 22 C5 33   502         LD   (POINTE),HL
334C D1         503         POP  DE
334D C1         504         POP  BC
334E C9         505         RET
334F            506 ;
334F            507 #COUNT
334F 01 01 00   508         LD   BC,1
3352            509 COT1
3352 FE 00      510         CP   0
3354 28 05      511         JR   Z,COT2
3356 03         512         INC  BC
3357 13         513         INC  DE
3358 1A         514         LD   A,(DE)
3359 18 F7      515         JR   COT1
335B            516 COT2
335B 3E 0D      517         LD   A,$0D
335D 12         518         LD   (DE),A
335E C9         519         RET
335F            520 ;
335F            521 #SEARCH
335F 01 00 00   522         LD   BC,0
3362 ED 5B BF   523         LD   DE,(TEXTS)
3365 33
3366            524 SR1
3366 1A         525         LD   A,(DE)
3367 FE 00      526         CP   0
3369 28 08      527         JR   Z,SR2
336B 13         528         INC  DE
336C FE 0D      529         CP   $0D
336E 20 F6      530         JR   NZ,SR1
3370 03         531         INC  BC
3371 18 F3      532         JR   SR1
3373            533 SR2
3373 ED 53 C1   534         LD   (TEXTE),DE
3376 33
3377 ED 43 C5   535         LD   (POINTE),BC
337A 33
337B C9         536         RET
337C            537 TEXTSKIP
337C CD A4 33   538         CALL #HLDECI
337F 38 34      539         JR   C,ERR
3381            540 TEXTSKIP1
3381 ED 5B C5   541         LD   DE,(POINTE)
3384 33
3385 EB         542         EX   DE,HL
3386 B7         543         OR   A
3387 ED 52      544         SBC  HL,DE
3389 EB         545         EX   DE,HL
338A 38 29      546         JR   C,ERR
338C 7C         547         LD   A,H
338D B5         548         OR   L
338E 28 25      549         JR   Z,ERR
3390            550 ;
3390 E5         551         PUSH HL
3391 ED 5B BF   552         LD   DE,(TEXTS)
3394 33
3395            553 TS1
3395 2B         554         DEC  HL
3396 7C         555         LD   A,H
3397 B5         556         OR   L
3398 20 02      557         JR   NZ,TS2
339A E1         558         POP  HL
339B C8         559         RET  Z
339C            560 TS2
339C 1A         561         LD   A,(DE)
339D FE 0D      562         CP   $0D
339F 13         563         INC  DE
33A0 20 FA      564         JR   NZ,TS2
33A2 18 F1      565         JR   TS1
33A4            566 #HLDECI
33A4 D5         567         PUSH DE
33A5 DD E1      568         POP  IX
33A7            569 #HLDECI_1
33A7 CD 5D 3F   570         CALL SPC_SKIP
33AA CD 6B 3F   571         CALL DECI
33AD 7C         572         LD   A,H
33AE B5         573         OR   L
33AF C0         574         RET  NZ
33B0 37         575         SCF
33B1 C9         576         RET
33B2            577 ;
33B2            578 ERRPRT
33B2 CD 33 20   579         CALL #ERROR
33B5            580 ERR
33B5 ED 7B C8   581         LD   SP,(WRK)
33B8 33
33B9 CD C4 1F   582         CALL #BELL
33BC C3 31 30   583         JP   E_MAIN
33BF            584 TEXTS
33BF 00 4E      585         DW   $4E00
33C1            586 TEXTE
33C1 00 4E      587         DW   $4E00
33C3            588 SADDR
33C3 00 4E      589         DW   $4E00
33C5            590 POINTE
33C5 00 00      591         DW   0
33C7            592 LPFLAG
33C7 00         593         DB   0
33C8            594 WRK
33C8 00 00      595         DW   0
33CA            596 P_WRK
33CA 00 00      597         DW   0
33CC            598 @PRTON
33CC 4F 4E      599         DM   "ON"
33CE 00         600         DB   0
33CF            601 @PRTOFF
33CF 4F 46 46   602         DM   "OFF"
33D2 00         603         DB   0
33D3            604 CRUN
33D3 CD B2 1F   605         CALL #HLHEX
33D6 DA B5 33   606         JP   C,ERR
33D9 22 E0 46   607         LD   (CCTEXT),HL
33DC E5         608         PUSH HL
33DD DD E1      609         POP  IX
33DF 3E 01      610         LD   A,1
33E1 18 05      611         JR   RUN1
33E3            612 ;
33E3            613 ; STACK Interpreter ver 1.0
33E3            614 ;
33E3            615 RUN
33E3 DD 2A BF   616         LD   IX,(TEXTS)
33E6 33
33E7            617 DIRECT
33E7 AF         618         XOR  A
33E8            619 RUN1
33E8 ED 73 CB   620         LD   (CPUSTK),SP
33EB 46
33EC 32 DF 46   621         LD   (CCFLAG),A
33EF            622 ;
33EF            623 ; LABEL TABLE CLEAR
33EF            624 ;
33EF            625 LABELCLR
33EF 21 EE 49   626         LD   HL,LABELT
33F2 11 EF 49   627         LD   DE,LABELT+1
33F5 01 FF 03   628         LD   BC,1023
33F8 36 00      629         LD   (HL),0
33FA ED B0      630         LDIR
33FC            631 ;
33FC            632 ; STACK SYOKIKA
33FC            633 ;
33FC 21 00 AE   634         LD   HL,@RET_SP
33FF 22 E2 46   635         LD   (RET_SP),HL
3402 21 00 AA   636         LD   HL,@VAR_SP
3405 F9         637         LD   SP,HL
3406            638 ;
3406            639 ; START
3406            640 ;
```


▶「イースIII」をしたら２日で解けた。ううっ，悲しい。
石井　敏雄（17）X68000　京都府

```
3406            641 MAIN
3406 3A DF 46   642          LD   A,(CCFLAG)
3409 B7         643          OR   A
340A C2 E3 42   644          JP   NZ,CCMAIN
340D            645 ;
340D 21 00 00   646          LD   HL,0
3410 39         647          ADD  HL,SP
3411 01 01 AA   648          LD   BC,@VAR_SP+1
3414 B7         649          OR   A
3415 ED 42      650          SBC  HL,BC
3417 D2 28 40   651          JP   NC,ERROR2
341A CD CD 1F   652          CALL #BREAK
341D CA 52 40   653          JP   Z,STOP
3420 CD 5D 3F   654          CALL SPC_SKIP
3423 DD 7E 00   655          LD   A,(IX)
3426 4F         656          LD   C,A
3427 DD 23      657          INC  IX
3429 B7         658          OR   A
342A CA 07 41   659          JP   Z,END
342D FE 0D      660          CP   $0D
342F 28 D5      661          JR   Z,MAIN
3431            662 ;
3431 FE 40      663          CP   '@'
3433 30 67      664          JR   NC,KAISEKI
3435 FE 25      665          CP   '%'
3437 CA D5 34   666          JP   Z,LABEL
343A FE 3B      667          CP   ';'
343C CA EF 34   668          JP   Z,REM
343F FE 22      669          CP   '"'
3441 CA 01 35   670          JP   Z,STRING
3444 FE 23      671          CP   '#'
3446 CA 07 35   672          JP   Z,HEN
3449 FE 2E      673          CP   '.'
344B CA 29 35   674          JP   Z,LET
344E FE 24      675          CP   '$'
3450 CA 7E 35   676          JP   Z,HEX
3453            677 ;
3453 FE 2D      678          CP   '-'
3455 28 14      679          JR   Z,MINUS
3457 FE 27      680          CP   "'"
3459 28 25      681          JR   Z,LSUUJI
345B FE 30      682          CP   '0'
345D 38 3D      683          JR   C,KAISEKI
345F FE 3A      684          CP   '9'+1
3461 30 39      685          JR   NC,KAISEKI
3463 DD 2B      686          DEC  IX
3465 CD 6B 3F   687          CALL DECI
3468 E5         688          PUSH HL
3469 18 9B      689          JR   MAIN
346B            690 MINUS
346B DD 7E 00   691          LD   A,(IX)
346E FE 21      692          CP   '!'
3470 38 09      693          JR   C,MINUS1
3472 CD 6B 3F   694          CALL DECI
3475 CD EC 39   695          CALL NEGATE
3478 E5         696          PUSH HL
3479 18 8B      697          JR   MAIN
347B            698 MINUS1
347B 06 2D      699          LD   B,'-'
347D C3 B1 34   700          JP   K_SKIP
3480            701 LSUUJI
3480 DD 7E 00   702          LD   A,(IX)
3483 FE 2D      703          CP   '-'
3485 28 08      704          JR   Z,LSUUJI1
3487 CD 89 3F   705          CALL HLDEDECI
348A E5         706          PUSH HL
348B D5         707          PUSH DE
348C C3 06 34   708          JP   MAIN
348F            709 LSUUJI1
348F DD 23      710          INC  IX
3491 CD 89 3F   711          CALL HLDEDECI
3494 CD F4 39   712          CALL NEGATE2
3497 E5         713          PUSH HL
3498 D5         714          PUSH DE
3499 C3 06 34   715          JP   MAIN
349C            716 KAISEKI
349C 47         717          LD   B,A
349D DD 7E 00   718          LD   A,(IX)
34A0 DD 23      719          INC  IX
34A2 FE 20      720          CP   ' '
34A4 28 0B      721          JR   Z,K_SKIP
34A6 FE 0D      722          CP   $0D
34A8 28 07      723          JR   Z,K_SKIP
34AA FE 09      724          CP   $09       ;TABCODE
34AC 28 03      725          JR   Z,K_SKIP
34AE 80         726          ADD  A,B
34AF 18 EB      727          JR   KAISEKI
34B1            728 K_SKIP
34B1 78         729          LD   A,B
34B2 21 19 44   730          LD   HL,COM_TBL
34B5 85         731          ADD  A,L
34B6 30 01      732          JR   NC,K_SKIP1
34B8 24         733          INC  H
34B9            734 K_SKIP1
34B9 6F         735          LD   L,A
34BA 7E         736          LD   A,(HL)
34BB 5F         737          LD   E,A
34BC 21 43 46   738          LD   HL,TOPDATA
34BF 85         739          ADD  A,L
34C0 30 01      740          JR   NC,K_SKIP2
34C2 24         741          INC  H
34C3            742 K_SKIP2
34C3 6F         743          LD   L,A
34C4 7E         744          LD   A,(HL)
34C5 B9         745          CP   C
34C6 C2 22 40   746          JP   NZ,ERROR1
34C9            747 ;
34C9 16 00      748          LD   D,0
34CB 21 19 45   749          LD   HL,JUMPTBL
34CE 19         750          ADD  HL,DE
34CF 19         751          ADD  HL,DE
34D0 5E         752          LD   E,(HL)
34D1 23         753          INC  HL
34D2 56         754          LD   D,(HL)
34D3 EB         755          EX   DE,HL
34D4 E9         756          JP   (HL)
34D5            757 LABEL
34D5 CD 6B 3F   758          CALL DECI
34D8 7C         759          LD   A,H
34D9 FE 02      760          CP   2
34DB 30 0A      761          JR   NC,LABEL1'
34DD            762 LABEL1
34DD 29         763          ADD  HL,HL
34DE DD E5      764          PUSH IX
34E0 C1         765          POP  BC
34E1 CD 0F 39   766          CALL #LABELW
34E4 C3 06 34   767          JP   MAIN
34E7            768 LABEL1'
34E7 FE 08      769          CP   8
34E9 D2 4C 40   770          JP   NC,ERROR5
34EC C3 06 34   771          JP   MAIN
34EF            772 REM
34EF 06 00      773          LD   B,0
34F1            774 REM1
34F1 DD 7E 00   775          LD   A,(IX)
34F4 DD 23      776          INC  IX
34F6 FE 0D      777          CP   $0D
34F8 CA 06 34   778          JP   Z,MAIN
34FB B8         779          CP   B
34FC CA 06 34   780          JP   Z,MAIN
34FF 18 F0      781          JR   REM1
3501            782 STRING
3501 DD E5      783          PUSH IX
3503 06 22      784          LD   B,'"'
3505 18 EA      785          JR   REM1
3507            786 HEN
3507 DD 7E 00   787          LD   A,(IX)
350A FE 23      788          CP   "#"
350C 28 0A      789          JR   Z,HEN2
350E CD 4B 35   790          CALL VAR_ADRS
3511 5E         791          LD   E,(HL)
3512 23         792          INC  HL
3513 56         793          LD   D,(HL)
3514 D5         794          PUSH DE
3515 C3 06 34   795          JP   MAIN
3518            796 HEN2
3518 DD 23      797          INC  IX
351A CD 4B 35   798          CALL VAR_ADRS
351D 5E         799          LD   E,(HL)
351E 23         800          INC  HL
351F 56         801          LD   D,(HL)
3520 D5         802          PUSH DE
3521 23         803          INC  HL
3522 5E         804          LD   E,(HL)
3523 23         805          INC  HL
3524 56         806          LD   D,(HL)
3525 D5         807          PUSH DE
3526 C3 06 34   808          JP   MAIN
3529            809 LET
3529 DD 7E 00   810          LD   A,(IX)
352C FE 2E      811          CP   '.'
352E 28 0A      812          JR   Z,LET2
3530 CD 4B 35   813          CALL VAR_ADRS
3533 D1         814          POP  DE
3534 73         815          LD   (HL),E
3535 23         816          INC  HL
3536 72         817          LD   (HL),D
3537 C3 06 34   818          JP   MAIN
353A            819 LET2
353A DD 23      820          INC  IX
353C CD 4B 35   821          CALL VAR_ADRS
353F D1         822          POP  DE
3540 C1         823          POP  BC
3541 71         824          LD   (HL),C
3542 23         825          INC  HL
3543 70         826          LD   (HL),B
3544 23         827          INC  HL
3545 73         828          LD   (HL),E
3546 23         829          INC  HL
3547 72         830          LD   (HL),D
3548 C3 06 34   831          JP   MAIN
354B            832 VAR_ADRS
354B DD 7E 00   833          LD   A,(IX)
354E DD 23      834          INC  IX
3550 FE 41      835          CP   'A'
3552 DA 22 40   836          JP   C,ERROR1
3555 FE 5B      837          CP   "Z"+1     ;
3557 D2 22 40   838          JP   NC,ERROR1 ;
355A D6 41      839          SUB  'A'
355C 87         840          ADD  A,A
355D 87         841          ADD  A,A
355E 26 00      842          LD   H,0
3560 6F         843          LD   L,A
3561 44         844          LD   B,H
3562 4D         845          LD   C,L
3563 29         846          ADD  HL,HL
3564 29         847          ADD  HL,HL
3565 09         848          ADD  HL,BC
3566 01 E6 47   849          LD   BC,VAR
3569 09         850          ADD  HL,BC
356A            851 ;
356A DD 7E 00   852          LD   A,(IX)
356D FE 31      853          CP   '1'
356F D8         854          RET  C
3570 FE 3A      855          CP   '9'+1
3572 D0         856          RET  NC
3573 DD 23      857          INC  IX
3575 D6 30      858          SUB  '0'
3577 87         859          ADD  A,A
3578 85         860          ADD  A,L
3579 30 01      861          JR   NC,V_SKIP
357B 24         862          INC  H
357C            863 V_SKIP
357C 6F         864          LD   L,A
357D C9         865          RET
357E            866 HEX
357E CD 85 35   867          CALL #HEX
3581 E5         868          PUSH HL
3582 C3 06 34   869          JP   MAIN
3585            870 ;
3585            871 #HEX
3585 21 00 00   872          LD   HL,0
3588            873 HEX1
3588 DD 7E 00   874          LD   A,(IX)
358B FE 30      875          CP   "0"
358D D8         876          RET  C
358E FE 47      877          CP   "F"+1
3590 D0         878          RET  NC
3591            879 ;
3591 FE 3A      880          CP   "9"+1
3593 38 05      881          JR   C,HEX2
3595 FE 41      882          CP   "A"
3597 D8         883          RET  C
3598 D6 07      884          SUB  7
359A            885 HEX2
359A D6 30      886          SUB  30H
359C 29         887          ADD  HL,HL
359D 29         888          ADD  HL,HL
359E 29         889          ADD  HL,HL
359F 29         890          ADD  HL,HL
35A0 85         891          ADD  A,L
35A1 30 01      892          JR   NC,HEXSKIP
35A3 24         893          INC  H
35A4            894 HEXSKIP
35A4 6F         895          LD   L,A
35A5 DD 23      896          INC   IX
35A7 18 DF      897          JR   HEX1
35A9            898 ;
35A9            899 ; ｴﾝｻﾞﾝ (1)
35A9            900 ;
35A9            901 @TASU
35A9 D1         902          POP  DE
35AA E1         903          POP  HL
35AB 19         904          ADD  HL,DE
35AC E5         905          PUSH HL
35AD C3 06 34   906          JP   MAIN
35B0            907 @HIKU
35B0 D1         908          POP  DE
35B1 E1         909          POP  HL
35B2 B7         910          OR   A
35B3 ED 52      911          SBC  HL,DE
35B5 E5         912          PUSH HL
35B6 C3 06 34   913          JP   MAIN
35B9            914 @MLT
35B9 D1         915          POP  DE
35BA E1         916          POP  HL
35BB CD 33 36   917          CALL MLT
35BE E5         918          PUSH HL
35BF C3 06 34   919          JP   MAIN
35C2            920 @DIV
35C2 D1         921          POP  DE
35C3 E1         922          POP  HL
35C4 CD 46 36   923          CALL DIV
35C7 E5         924          PUSH HL
35C8 C3 06 34   925          JP   MAIN
35CB            926 @MOD
35CB D1         927          POP  DE
35CC E1         928          POP  HL
35CD CD 46 36   929          CALL DIV
35D0 D5         930          PUSH DE
35D1 C3 06 34   931          JP   MAIN
35D4            932 @DIVMOD
35D4 D1         933          POP  DE
35D5 E1         934          POP  HL
35D6 CD 46 36   935          CALL DIV
35D9 E5         936          PUSH HL
35DA D5         937          PUSH DE
35DB C3 06 34   938          JP   MAIN
35DE            939 @==
35DE E1         940          POP  HL
35DF D1         941          POP  DE
35E0 B7         942          OR   A
35E1 ED 52      943          SBC  HL,DE
35E3 21 00 00   944          LD   HL,0
35E6 20 01      945          JR   NZ,@==1
35E8 23         946          INC  HL
35E9            947 @==1
35E9 E5         948          PUSH HL
35EA C3 06 34   949          JP   MAIN
35ED            950 @<
35ED D1         951          POP  DE
35EE E1         952          POP  HL
35EF            953 @<'
35EF B7         954          OR   A
35F0 ED 52      955          SBC  HL,DE
35F2 21 01 00   956          LD   HL,1
35F5 38 01      957          JR   C,@<1
35F7 2B         958          DEC  HL
35F8            959 @<1
35F8 E5         960          PUSH HL
35F9 C3 06 34   961          JP   MAIN
35FC            962 @>
35FC E1         963          POP  HL
35FD D1         964          POP  DE
35FE 18 EF      965          JR   @<'
3600            966 @!=
3600 E1         967          POP  HL
3601 D1         968          POP  DE
3602 B7         969          OR   A
3603 ED 52      970          SBC  HL,DE
3605 21 01 00   971          LD   HL,1
3608 20 01      972          JR   NZ,@!=1
360A 2B         973          DEC  HL
360B            974 @!=1
360B E5         975          PUSH HL
360C C3 06 34   976          JP   MAIN
360F            977 @AND
360F D1         978          POP  DE
3610 E1         979          POP  HL
3611 7D         980          LD   A,L
3612 A3         981          AND  E
3613 6F         982          LD   L,A
3614 7C         983          LD   A,H
3615 A2         984          AND  D
3616 67         985          LD   H,A
3617 E5         986          PUSH HL
3618 C3 06 34   987          JP   MAIN
361B            988 @OR
361B D1         989          POP  DE
361C E1         990          POP  HL
361D 7D         991          LD   A,L
361E B3         992          OR   E
361F 6F         993          LD   L,A
3620 7C         994          LD   A,H
3621 B2         995          OR   D
3622 67         996          LD   H,A
3623 E5         997          PUSH HL
3624 C3 06 34   998          JP   MAIN
3627            999 @XOR
3627 D1        1000          POP  DE
3628 E1        1001          POP  HL
3629 7D        1002          LD   A,L
362A AB        1003          XOR  E
362B 6F        1004          LD   L,A
362C 7C        1005          LD   A,H
362D AA        1006          XOR  D
362E 67        1007          LD   H,A
362F E5        1008          PUSH HL
3630 C3 06 34  1009          JP   MAIN
3633           1010 ;
3633           1011 ; HL=HL*DE
3633           1012 ;
3633           1013 MLT
3633 4D        1014          LD   C,L
3634 44        1015          LD   B,H
3635 3E 10     1016          LD   A,16
3637 21 00 00  1017          LD   HL,0
363A           1018 MLT1
363A 29        1019          ADD  HL,HL
363B CB 23     1020          SLA  E
363D CB 12     1021          RL   D
363F 30 01     1022          JR   NC,MLT2
3641 09        1023          ADD  HL,BC
3642           1024 MLT2
3642 3D        1025          DEC  A
3643 20 F5     1026          JR   NZ,MLT1
3645 C9        1027          RET
3646           1028 ;
3646           1029 ; HL=HL/DE
3646           1030 ;
3646           1031 ; DE=HL mod DE
3646           1032 ;
3646           1033 DIV
3646 4B        1034          LD   C,E
3647 42        1035          LD   B,D
3648 5D        1036          LD   E,L
3649 54        1037          LD   D,H
364A 3E 10     1038          LD   A,16
364C 21 00 00  1039          LD   HL,0
364F           1040 DIV1
364F CB 23     1041          SLA  E
3651 CB 12     1042          RL   D
3653 ED 6A     1043          ADC  HL,HL
3655 E5        1044          PUSH HL
3656 B7        1045          OR   A
3657 ED 42     1046          SBC  HL,BC
3659 E1        1047          POP  HL
365A 38 03     1048          JR   C,DIV2
365C ED 42     1049          SBC  HL,BC
365E 13        1050          INC  DE
365F           1051 DIV2
365F 3D        1052          DEC  A
3660 20 ED     1053          JR   NZ,DIV1
3662 EB        1054          EX   DE,HL
3663 C9        1055          RET
3664           1056 ;
3664           1057 ; STACK ｿｳｻ
3664           1058 ;
3664           1059 @DROP
3664 E1        1060          POP  HL
3665 C3 06 34  1061          JP   MAIN
3668           1062 @SWAP1
3668 D1        1063          POP  DE
3669 E1        1064          POP  HL
366A D5        1065          PUSH DE
366B E5        1066          PUSH HL
366C C3 06 34  1067          JP   MAIN
366F           1068 @COPY
366F E1        1069          POP  HL
3670 E5        1070          PUSH HL
3671 E5        1071          PUSH HL
3672 C3 06 34  1072          JP   MAIN
3675           1073 @ROT
3675 E1        1074          POP  HL
3676 D1        1075          POP  DE
3677 C1        1076          POP  BC
3678 D5        1077          PUSH DE
3679 E5        1078          PUSH HL
367A C5        1079          PUSH BC
367B C3 06 34  1080          JP   MAIN
367E           1081 ;
367E           1082 ; FUNCITON 1
367E           1083 ;
367E           1084 @KEY
367E CD CA 1F  1085          CALL #INKEY
3681 6F        1086          LD   L,A
3682 26 00     1087          LD   H,0
3684 E5        1088          PUSH HL
3685 C3 06 34  1089          JP   MAIN
3688           1090 @GETKEY
3688 CD D0 1F  1091          CALL #GETKY
368B 6F        1092          LD   L,A
368C 26 00     1093          LD   H,0
368E E5        1094          PUSH HL
368F C3 06 34  1095          JP   MAIN
3692           1096 @FLGET
3692 CD 21 20  1097          CALL #FLGET
3695 6F        1098          LD   L,A
3696 26 00     1099          LD   H,0
3698 E5        1100          PUSH HL
3699 C3 06 34  1101          JP   MAIN
369C           1102 @RND
369C 2A D3 46  1103          LD   HL,(RND0)
369F 54 5D     1104          LD   DE,HL
36A1 19        1105          ADD  HL,DE
36A2 19        1106          ADD  HL,DE
36A3 7D        1107          LD   A,L
36A4 84        1108          ADD  A,H
36A5 67        1109          LD   H,A
36A6 85        1110          ADD  A,L
36A7 6F        1111          LD   L,A
36A8 11 54 00  1112          LD   DE,$54
36AB 19        1113          ADD  HL,DE
36AC 22 D3 46  1114          LD   (RND0),HL
36AF E5        1115          PUSH HL
36B0 C3 06 34  1116          JP   MAIN
36B3           1117 @SCRN
36B3 D1        1118          POP  DE
36B4 E1        1119          POP  HL
36B5 63        1120          LD   H,E
36B6 CD 1B 20  1121          CALL #SCRN
36B9 6F        1122          LD   L,A
36BA 26 00     1123          LD   H,0
36BC E5        1124          PUSH HL
36BD C3 06 34  1125          JP   MAIN
36C0           1126 ;
```


▶ゲーム業界というのは，夢を与える仕事であると考える。ゆえに，もうけだけをものさしにしてほしくない。何がいいたいかというと,「BPSさん，アーコン出して！」「テレネットさん，ヴァリスIIをX1で……」ちょっとムリか。優子ちゃんがPGSでしゃべったりして……。

西野　裕(22)X1/turboII　京都府

```
36C0           1127 ; PRINT 1
36C0           1128 ;
36C0           1129 @HEX2
36C0 E1        1130         POP  HL
36C1 7D        1131         LD   A,L
36C2 CD C1 1F  1132         CALL #PRTHX
36C5 C3 06 34  1133         JP   MAIN
36C8           1134 @HEX4
36C8 E1        1135         POP  HL
36C9 CD BE 1F  1136         CALL #PRTHL
36CC C3 06 34  1137         JP   MAIN
36CF           1138 @PRINT
36CF E1        1139         POP  HL
36D0 CD 0E 40  1140         CALL @DEC2
36D3 C3 06 34  1141         JP   MAIN
36D6           1142 @CHR
36D6 E1        1143         POP  HL
36D7 7D        1144         LD   A,L
36D8 CD F4 1F  1145         CALL #PRINT
36DB C3 06 34  1146         JP   MAIN
36DE           1147 @PRTS
36DE E1        1148         POP  HL
36DF           1149 @PRTS1
36DF 7E        1150         LD   A,(HL)
36E0 FE 22     1151         CP   '"'
36E2 CA 06 34  1152         JP   Z,MAIN
36E5 FE 0D     1153         CP   $0D
36E7 CA 06 34  1154         JP   Z,MAIN
36EA CD F4 1F  1155         CALL #PRINT
36ED 23        1156         INC  HL
36EE 18 EF     1157         JR   @PRTS1
36F0           1158 @COTR
36F0 E1        1159         POP  HL
36F1           1160 @COTR1
36F1 7E        1161         LD   A,(HL)
36F2 FE 22     1162         CP   '"'
36F4 CA 06 34  1163         JP   Z,MAIN
36F7 FE 0D     1164         CP   $0D
36F9 CA 06 34  1165         JP   Z,MAIN
36FC 0E 1C     1166         LD   C,$1C
36FE FE 52     1167         CP   'R'
3700 28 1B     1168         JR   Z,@COTR2
3702 0C        1169         INC  C
3703 FE 4C     1170         CP   'L'
3705 28 16     1171         JR   Z,@COTR2
3707 0C        1172         INC  C
3708 FE 55     1173         CP   'U'
370A 28 11     1174         JR   Z,@COTR2
370C 0C        1175         INC  C
370D FE 44     1176         CP   'D'
370F 28 0C     1177         JR   Z,@COTR2
3711 0E 0C     1178         LD   C,$0C
3713 FE 43     1179         CP   'C'
3715 28 06     1180         JR   Z,@COTR2
3717 0C        1181         INC  C
3718 FE 2F     1182         CP   '/'
371A C2 22 40  1183         JP   NZ,ERROR1
371D           1184 @COTR2
371D 79        1185         LD   A,C
371E CD F4 1F  1186         CALL #PRINT
3721 23        1187         INC  HL
3722 18 CD     1188         JR   @COTR1
3724           1189 @CR
3724 3E 0D     1190         LD   A,$0D
3726 CD F4 1F  1191         CALL #PRINT
3729 C3 06 34  1192         JP   MAIN
372C           1193 ;
372C           1194 ; ｿﾉﾀ 1
372C           1195 ;
372C           1196 @INC
372C CD 5D 3F  1197         CALL SPC_SKIP
372F CD 4B 35  1198         CALL VAR_ADRS
3732           1199 @INC1
3732 5E        1200         LD   E,(HL)
3733 23        1201         INC  HL
3734 56        1202         LD   D,(HL)
3735 13        1203         INC  DE
3736 72        1204         LD   (HL),D
3737 2B        1205         DEC  HL
3738 73        1206         LD   (HL),E
3739 C3 06 34  1207         JP   MAIN
373C           1208 @DEC
373C CD 5D 3F  1209         CALL SPC_SKIP
373F CD 4B 35  1210         CALL VAR_ADRS
3742           1211 @DEC1
3742 5E        1212         LD   E,(HL)
3743 23        1213         INC  HL
3744 56        1214         LD   D,(HL)
3745 1B        1215         DEC  DE
3746 72        1216         LD   (HL),D
3747 2B        1217         DEC  HL
3748 73        1218         LD   (HL),E
3749 C3 06 34  1219         JP   MAIN
374C           1220 @WIDCH
374C E1        1221         POP  HL
374D 7D        1222         LD   A,L
374E CD 30 20  1223         CALL #WIDCH
3751 C3 06 34  1224         JP   MAIN
3754           1225 @BELL
3754 E1        1226         POP  HL
3755 45        1227         LD   B,L
3756 78        1228         LD   A,B
3757 B7        1229         OR   A
3758 CA 06 34  1230         JP   Z,MAIN
375B           1231 @BELL1
375B CD C4 1F  1232         CALL #BELL
375E 10 FB     1233         DJNZ @BELL1
3760 C3 06 34  1234         JP   MAIN
3763           1235 @LOCATE
3763 D1        1236         POP  DE
3764 E1        1237         POP  HL
3765 63        1238         LD   H,E
3766 CD 1E 20  1239         CALL #LOC
3769 C3 06 34  1240         JP   MAIN
376C           1241 ;
376C           1242 ; ｾｲｷﾞｮｹｲ 1
376C           1243 ;
376C           1244 @REPEAT
376C ED 73 E4  1245         LD   (STK_WR),SP
376F 46
3770 ED 7B E2  1246         LD   SP,(RET_SP)
3773 46
3774 DD E5     1247         PUSH IX
3776 ED 73 E2  1248         LD   (RET_SP),SP
3779 46
377A ED 7B E4  1249         LD   SP,(STK_WR)
377D 46
377E           1250 @REPEAT1
377E C3 06 34  1251         JP   MAIN
3781           1252 @UNTIL
3781 D1        1253         POP  DE
3782 2A E2 46  1254         LD   HL,(RET_SP)
3785 01 00 AE  1255         LD   BC,@RET_SP
3788 B7        1256         OR   A
3789 ED 42     1257         SBC  HL,BC
378B D2 2E 40  1258         JP   NC,ERROR3
378E 7A        1259         LD   A,D
378F B3        1260         OR   E
3790 CA 9D 37  1261         JP   Z,UNTIL1
3793 2A E2 46  1262         LD   HL,(RET_SP)
3796 23        1263         INC  HL
3797 23        1264         INC  HL
3798 22 E2 46  1265         LD   (RET_SP),HL
379B 18 E1     1266         JR   @REPEAT1
379D           1267 UNTIL1
379D 2A E2 46  1268         LD   HL,(RET_SP)
37A0 5E        1269         LD   E,(HL)
37A1 23        1270         INC  HL
37A2 56        1271         LD   D,(HL)
37A3 D5        1272         PUSH DE
37A4 DD E1     1273         POP  IX
37A6 18 D6     1274         JR   @REPEAT1
37A8           1275 @GOTO
37A8 CD 5D 3F  1276         CALL SPC_SKIP
37AB CD 6B 3F  1277         CALL DECI
37AE           1278 GOTO
37AE 7C        1279         LD   A,H
37AF FE 02     1280         CP   2
37B1 30 0F     1281         JR   NC,GOTO'
37B3 29        1282         ADD  HL,HL
37B4 CD 03 39  1283         CALL #LABELR
37B7 78        1284         LD   A,B
37B8 B1        1285         OR   C
37B9 CA C8 37  1286         JP   Z,GOTO1
37BC C5        1287         PUSH BC
37BD DD E1     1288         POP  IX
37BF C3 06 34  1289         JP   MAIN
37C2           1290 GOTO'
37C2 FE 08     1291         CP   8
37C4 D2 4C 40  1292         JP   NC,ERROR5
37C7 29        1293         ADD  HL,HL
37C8           1294 GOTO1
37C8 DD 22 CF  1295         LD   (LINE_WR1),IX
37CB 46
37CC AF        1296         XOR  A
37CD CB 3C     1297         SRL  H                   ;HL=HL/2
37CF CB 1D     1298         RR   L
37D1 22 CD 46  1299         LD   (LINE_WR),HL
37D4 DD 2A BF  1300         LD   IX,(TEXTS)
37D7 33
37D8           1301 GOTO2
37D8 DD 7E 00  1302         LD   A,(IX)
37DB B7        1303         OR   A
37DC 28 22     1304         JR   Z,G_ERR
37DE FE 25     1305         CP   '%'
37E0 C2 F5 37  1306         JP   NZ,G_SKIP
37E3 DD 23     1307         INC  IX
37E5 CD 6B 3F  1308         CALL DECI
37E8 ED 5B CD  1309         LD   DE,(LINE_WR)
37EB 46
37EC B7        1310         OR   A
37ED ED 52     1311         SBC  HL,DE
37EF 20 04     1312         JR   NZ,G_SKIP
37F1 19        1313         ADD  HL,DE
37F2 C3 DD 34  1314         JP   LABEL1
37F5           1315 G_SKIP
37F5 DD 7E 00  1316         LD   A,(IX)
37F8 DD 23     1317         INC  IX
37FA FE 0D     1318         CP   $0D
37FC 28 DA     1319         JR   Z,GOTO2
37FE 18 F5     1320         JR   G_SKIP
3800           1321 G_ERR
3800 DD 2A CF  1322         LD   IX,(LINE_WR1)
3803 46
3804 C3 34 40  1323         JP   ERROR4
3807           1324 @IF
3807 E1        1325         POP  HL
3808 7D        1326         LD   A,L
3809 B4        1327         OR   H
380A 28 03     1328         JR   Z,IF_SKIP
380C C3 06 34  1329         JP   MAIN
380F           1330 IF_SKIP
380F DD 7E 00  1331         LD   A,(IX)
3812 DD 23     1332         INC  IX
3814 FE 0D     1333         CP   $0D
3816 CA 06 34  1334         JP   Z,MAIN
3819 18 F4     1335         JR   IF_SKIP
381B           1336 @GOSUB
381B CD 5D 3F  1337         CALL SPC_SKIP
381E CD 6B 3F  1338         CALL DECI
3821           1339 @GOSUB1
3821 ED 73 E4  1340         LD   (STK_WR),SP
3824 46
3825 ED 7B E2  1341         LD   SP,(RET_SP)
3828 46
3829 DD E5     1342         PUSH IX
382B ED 73 E2  1343         LD   (RET_SP),SP
382E 46
382F ED 7B E4  1344         LD   SP,(STK_WR)
3832 46
3833 C3 AE 37  1345         JP   GOTO
3836           1346 @RET
3836 2A E2 46  1347         LD   HL,(RET_SP)
3839 01 00 AE  1348         LD   BC,@RET_SP
383C B7        1349         OR   A
383D ED 42     1350         SBC  HL,BC
383F D2 2E 40  1351         JP   NC,ERROR3
3842 ED 73 E4  1352         LD   (STK_WR),SP
3845 46
3846 ED 7B E2  1353         LD   SP,(RET_SP)
3849 46
384A DD E1     1354         POP  IX
384C ED 73 E2  1355         LD   (RET_SP),SP
384F 46
3850 ED 7B E4  1356         LD   SP,(STK_WR)
3853 46
3854 C3 06 34  1357         JP   MAIN
3857           1358 @DO
3857 E1        1359         POP  HL
3858 D1        1360         POP  DE
3859 ED 73 E4  1361         LD   (STK_WR),SP
385C 46
385D ED 7B E2  1362         LD   SP,(RET_SP)
3860 46
3861 DD E5     1363         PUSH IX
3863 E5        1364         PUSH HL
3864 D5        1365         PUSH DE
3865 ED 73 E2  1366         LD   (RET_SP),SP
3868 46
3869 ED 7B E4  1367         LD   SP,(STK_WR)
386C 46
386D C3 06 34  1368         JP   MAIN
3870           1369 @LOOP!
3870 2A E2 46  1370         LD   HL,(RET_SP)
3873 01 FC AD  1371         LD   BC,@RET_SP-4
3876 B7        1372         OR   A
3877 ED 42     1373         SBC  HL,BC
3879 D2 2E 40  1374         JP   NC,ERROR3
387C ED 73 E4  1375         LD   (STK_WR),SP
387F 46
3880 ED 7B E2  1376         LD   SP,(RET_SP)
3883 46
3884 D1        1377         POP  DE
3885 E1        1378         POP  HL
3886 FD E1     1379         POP  IY
3888 13        1380         INC  DE
3889 B7        1381         OR   A
388A ED 52     1382         SBC  HL,DE
388C 38 14     1383         JR   C,@LOOP!1
388E 19        1384         ADD  HL,DE
388F FD E5     1385         PUSH IY
3891 E5        1386         PUSH HL
3892 D5        1387         PUSH DE
3893 ED 73 E2  1388         LD   (RET_SP),SP
3896 46
3897 ED 7B E4  1389         LD   SP,(STK_WR)
389A 46
389B FD E5     1390         PUSH IY
389D DD E1     1391         POP  IX
389F C3 06 34  1392         JP   MAIN
38A2           1393 @LOOP!1
38A2 ED 73 E2  1394         LD   (RET_SP),SP
38A5 46
38A6 ED 7B E4  1395         LD   SP,(STK_WR)
38A9 46
38AA C3 06 34  1396         JP   MAIN
38AD           1397 @I?
38AD 11 00 00  1398         LD   DE,0
38B0           1399 I?
38B0 2A E2 46  1400         LD   HL,(RET_SP)
38B3 19        1401         ADD  HL,DE
38B4 5E        1402         LD   E,(HL)
38B5 23        1403         INC  HL
38B6 56        1404         LD   D,(HL)
38B7 D5        1405         PUSH DE
38B8 C3 06 34  1406         JP   MAIN
38BB           1407 @J?
38BB 11 06 00  1408         LD   DE,6
38BE 18 F0     1409         JR   I?
38C0           1410 @TR
38C0 E1        1411         POP  HL
38C1 ED 73 E4  1412         LD   (STK_WR),SP
38C4 46
38C5 ED 7B E2  1413         LD   SP,(RET_SP)
38C8 46
38C9 E5        1414         PUSH HL
38CA ED 73 E2  1415         LD   (RET_SP),SP
38CD 46
38CE ED 7B E4  1416         LD   SP,(STK_WR)
38D1 46
38D2 C3 06 34  1417         JP   MAIN
38D5           1418 @FR
38D5 2A E2 46  1419         LD   HL,(RET_SP)
38D8 01 00 AE  1420         LD   BC,@RET_SP
38DB B7        1421         OR   A
38DC ED 42     1422         SBC  HL,BC
38DE D2 2E 40  1423         JP   NC,ERROR3
38E1 ED 73 E4  1424         LD   (STK_WR),SP
38E4 46
38E5 ED 7B E2  1425         LD   SP,(RET_SP)
38E8 46
38E9 E1        1426         POP  HL
38EA ED 73 E2  1427         LD   (RET_SP),SP
38ED 46
38EE ED 7B E4  1428         LD   SP,(STK_WR)
38F1 46
38F2 E5        1429         PUSH HL
38F3 C3 06 34  1430         JP   MAIN
38F6           1431 @LEA
38F6 2A E2 46  1432         LD   HL,(RET_SP)
38F9 5E        1433         LD   E,(HL)
38FA 23        1434         INC  HL
38FB 56        1435         LD   D,(HL)
38FC 23        1436         INC  HL
38FD 73        1437         LD   (HL),E
38FE 23        1438         INC  HL
38FF 72        1439         LD   (HL),D
3900 C3 06 34  1440         JP   MAIN
3903           1441 #LABELR
3903 D5        1442         PUSH DE
3904 E5        1443         PUSH HL
3905 11 EE 49  1444         LD   DE,LABELT
3908 19        1445         ADD  HL,DE
3909 4E        1446         LD   C,(HL)
390A 23        1447         INC  HL
390B 46        1448         LD   B,(HL)
390C E1        1449         POP  HL
390D D1        1450         POP  DE
390E C9        1451         RET
390F           1452 #LABELW
390F D5        1453         PUSH DE
3910 E5        1454         PUSH HL
3911 11 EE 49  1455         LD   DE,LABELT
3914 19        1456         ADD  HL,DE
3915 71        1457         LD   (HL),C
3916 23        1458         INC  HL
3917 70        1459         LD   (HL),B
3918 E1        1460         POP  HL
3919 D1        1461         POP  DE
391A C9        1462         RET
391B           1463 ;
391B           1464 ; ﾏｼﾝｺﾞ ｹｲ
391B           1465 ;
391B           1466 @CALL
391B C1        1467         POP  BC
391C 2A DA 46  1468         LD   HL,(#HL)
391F ED 5B DC  1469         LD   DE,(#DE)
3922 46
3923 3A DE 46  1470         LD   A,(#A)
3926 ED 43 2B  1471         LD   (@CALL1+1),BC
3929 39
392A           1472 @CALL1
392A CD 00 00  1473         DB   $CD,0,0
392D 22 DA 46  1474         LD   (#HL),HL
3930 ED 53 DC  1475         LD   (#DE),DE
3933 46
3934 32 DE 46  1476         LD   (#A),A
3937 C3 06 34  1477         JP   MAIN
393A           1478 @PUTA
393A E1        1479         POP  HL
393B 7D        1480         LD   A,L
393C 32 DE 46  1481         LD   (#A),A
393F C3 06 34  1482         JP   MAIN
3942           1483 @GETA
3942 3A DE 46  1484         LD   A,(#A)
3945 6F        1485         LD   L,A
3946 26 00     1486         LD   H,0
3948 E5        1487         PUSH HL
3949 C3 06 34  1488         JP   MAIN
394C           1489 @PUTD
394C D1        1490         POP  DE
394D ED 53 DC  1491         LD   (#DE),DE
3950 46
3951 C3 06 34  1492         JP   MAIN
3954           1493 @GETD
3954 ED 5B DC  1494         LD   DE,(#DE)
3957 46
3958 D5        1495         PUSH DE
3959 C3 06 34  1496         JP   MAIN
395C           1497 @PUTH
395C E1        1498         POP  HL
395D 22 DA 46  1499         LD   (#HL),HL
3960 C3 06 34  1500         JP   MAIN
3963           1501 @GETH
3963 2A DA 46  1502         LD   HL,(#HL)
3966 E5        1503         PUSH HL
3967 C3 06 34  1504         JP   MAIN
396A           1505 ;
396A           1506 ; ﾒﾓﾘ ｿｳｻ 1
396A           1507 ;
396A           1508 @PEEKB
396A E1        1509         POP  HL
396B 5E        1510         LD   E,(HL)
396C 16 00     1511         LD   D,0
396E D5        1512         PUSH DE
396F C3 06 34  1513         JP   MAIN
3972           1514 @PEEKW
3972 E1        1515         POP  HL
3973 5E        1516         LD   E,(HL)
3974 23        1517         INC  HL
3975 56        1518         LD   D,(HL)
3976 D5        1519         PUSH DE
3977 C3 06 34  1520         JP   MAIN
397A           1521 @POKEB
397A E1        1522         POP  HL
397B D1        1523         POP  DE
397C 73        1524         LD   (HL),E
397D C3 06 34  1525         JP   MAIN
3980           1526 @POKEW
3980 E1        1527         POP  HL
3981 D1        1528         POP  DE
3982 73        1529         LD   (HL),E
3983 23        1530         INC  HL
3984 72        1531         LD   (HL),D
3985 C3 06 34  1532         JP   MAIN
3988           1533 ;
3988           1534 ; I/O ｿｳｻ
3988           1535 ;
3988           1536 @IN
3988 C1        1537         POP  BC
3989 ED 58     1538         IN   E,(C)
398B 16 00     1539         LD   D,0
398D D5        1540         PUSH DE
398E C3 06 34  1541         JP   MAIN
3991           1542 @OUT
3991 E1        1543         POP  HL
3992 C1        1544         POP  BC
3993 ED 69     1545         OUT  (C),L
3995 C3 06 34  1546         JP   MAIN
3998           1547 ;
3998           1548 ;
3998           1549 @HIGH
3998 E1        1550         POP  HL
3999 6C        1551         LD   L,H
399A 26 00     1552         LD   H,0
399C E5        1553         PUSH HL
399D C3 06 34  1554         JP   MAIN
39A0           1555 @LOW
39A0 E1        1556         POP  HL
39A1 26 00     1557         LD   H,0
39A3 E5        1558         PUSH HL
39A4 C3 06 34  1559         JP   MAIN
39A7           1560 @EX
39A7 E1        1561         POP  HL
39A8 7D        1562         LD   A,L
39A9 6C        1563         LD   L,H
39AA 67        1564         LD   H,A
39AB E5        1565         PUSH HL
39AC C3 06 34  1566         JP   MAIN
39AF           1567 @NOT
39AF E1        1568         POP  HL
39B0 7D        1569         LD   A,L
39B1 2F        1570         CPL
39B2 6F        1571         LD   L,A
39B3 7C        1572         LD   A,H
39B4 2F        1573         CPL
```

THE SENTINEL

▶うちの X68000のハードディスクがふっとんでしまった。40M バイトが，PDS のファイルが，使い込んだ辞書が……。う一悲しい。やっぱ，バックアップはこまめにしなければ，と実感した。　　　　畑山　雅章 (16) X68000EXPERT-HD　京都府

```
39B5 67          1574          LD   H,A
39B6 E5          1575          PUSH HL
39B7 C3 06 34    1576          JP   MAIN
39BA             1577 @ROR
39BA C1          1578          POP  BC
39BB E1          1579          POP  HL
39BC 41          1580          LD   B,C
39BD             1581 ROR1
39BD CB 3C       1582          SRL  H
39BF CB 1D       1583          RR   L
39C1 10 FA       1584          DJNZ ROR1
39C3 E5          1585          PUSH HL
39C4 C3 06 34    1586          JP   MAIN
39C7             1587 @ROL
39C7 C1          1588          POP  BC
39C8 E1          1589          POP  HL
39C9 41          1590          LD   B,C
39CA             1591 ROL1
39CA 29          1592          ADD  HL,HL
39CB 10 FD       1593          DJNZ ROL1
39CD E5          1594          PUSH HL
39CE C3 06 34    1595          JP   MAIN
39D1             1596 @CURX
39D1 CD 18 20    1597          CALL #CSR
39D4 26 00       1598          LD   H,0
39D6 E5          1599          PUSH HL
39D7 C3 06 34    1600          JP   MAIN
39DA             1601 @CURY
39DA CD 18 20    1602          CALL #CSR
39DD 6C          1603          LD   L,H
39DE 26 00       1604          LD   H,0
39E0 E5          1605          PUSH HL
39E1 C3 06 34    1606          JP   MAIN
39E4             1607 @NEGATE
39E4 E1          1608          POP  HL
39E5 CD EC 39    1609          CALL NEGATE
39E8 E5          1610          PUSH HL
39E9 C3 06 34    1611          JP   MAIN
39EC             1612 NEGATE
39EC 7D          1613          LD   A,L
39ED 2F          1614          CPL
39EE 6F          1615          LD   L,A
39EF 7C          1616          LD   A,H
39F0 2F          1617          CPL
39F1 67          1618          LD   H,A
39F2 23          1619          INC  HL
39F3 C9          1620          RET
39F4             1621 NEGATE2
39F4 7B          1622          LD   A,E
39F5 2F          1623          CPL
39F6 5F          1624          LD   E,A
39F7 7A          1625          LD   A,D
39F8 2F          1626          CPL
39F9 57          1627          LD   D,A
39FA 7D          1628          LD   A,L
39FB 2F          1629          CPL
39FC 6F          1630          LD   L,A
39FD 7C          1631          LD   A,H
39FE 2F          1632          CPL
39FF 67          1633          LD   H,A
3A00 01 01 00    1634          LD   BC,1
3A03 EB          1635          EX   DE,HL
3A04 09          1636          ADD  HL,BC
3A05 EB          1637          EX   DE,HL
3A06 0B          1638          DEC  BC
3A07 ED 4A       1639          ADC  HL,BC
3A09 C9          1640          RET
3A0A             1641 NEXT
3A0A                1          ORG  NEXT
3A0A                2          OFFSET $B000
3A0A                3 ;
3A0A                4 ; STRING ｹｲ
3A0A                5 ;
3A0A                6 @STRCPY
3A0A E1             7          POP  HL
3A0B D1             8          POP  DE
3A0C                9 STRCPY1
3A0C 1A            10          LD   A,(DE)
3A0D 77            11          LD   (HL),A
3A0E FE 0D         12          CP   $0D
3A10 28 0A         13          JR   Z,STRCPY3
3A12 FE 22         14          CP   '"'
3A14 28 04         15          JR   Z.STRCPY2
3A16 23            16          INC  HL
3A17 13            17          INC  DE
3A18 18 F2         18          JR   STRCPY1
3A1A               19 STRCPY2
3A1A 36 0D         20          LD   (HL),$0D
3A1C               21 STRCPY3
3A1C C3 06 34      22          JP   MAIN
3A1F               23 @LEFT$
3A1F C1            24          POP  BC
3A20 E1            25          POP  HL
3A21 D1            26          POP  DE
3A22               27 LEFT1
3A22 1A            28          LD   A,(DE)
3A23 77            29          LD   (HL),A
3A24 FE 0D         30          CP   $0D
3A26 28 0D         31          JR   Z,LEFT3
3A28 FE 22         32          CP   '"'
3A2A 28 07         33          JR   Z,LEFT2
3A2C 23            34          INC  HL
3A2D 13            35          INC  DE
3A2E 0B            36          DEC  BC
3A2F 78            37          LD   A,B
3A30 B1            38          OR   C
3A31 20 EF         39          JR   NZ,LEFT1
3A33               40 LEFT2
3A33 36 0D         41          LD   (HL),$0D
3A35               42 LEFT3
3A35 C3 06 34      43          JP   MAIN
3A38               44 @RIGHT$
3A38 C1            45          POP  BC
3A39 FD E1         46          POP  IY
3A3B E1            47          POP  HL
3A3C 5D            48          LD   E,L
3A3D 54            49          LD   D,H
3A3E               50 RIGHT1
3A3E 7E            51          LD   A,(HL)
3A3F FE 22         52          CP   '"'
3A41 28 07         53          JR   Z,RIGHT2
3A43 FE 0D         54          CP   $0D
3A45 28 03         55          JR   Z,RIGHT2
3A47 23            56          INC  HL
3A48 18 F4         57          JR   RIGHT1
3A4A               58 RIGHT2
3A4A 2B            59          DEC  HL
3A4B E5            60          PUSH HL
3A4C B7            61          OR   A
3A4D ED 52         62          SBC  HL,DE
3A4F E1            63          POP  HL
3A50 28 05         64          JR   Z,RIGHT3
3A52 0B            65          DEC  BC
3A53 78            66          LD   A,B
3A54 B1            67          OR   C
3A55 20 F3         68          JR   NZ,RIGHT2
3A57               69 RIGHT3
3A57 FD E5         70          PUSH IY
3A59 D1            71          POP  DE
3A5A EB            72          EX   DE,HL
3A5B 18 AF         73          JR   STRCPY1
3A5D               74 @MID$
3A5D D9            75          EXX
3A5E C1            76          POP  BC
3A5F D9            77          EXX
3A60 C1            78          POP  BC
3A61 E1            79          POP  HL
3A62 D1            80          POP  DE
3A63               81 MID1
3A63 0B            82          DEC  BC
3A64 78            83          LD   A,B
3A65 B1            84          OR   C
3A66 28 0E         85          JR   Z,MID2
3A68 1A            86          LD   A,(DE)
3A69 FE 22         87          CP   '"'
3A6B CA 22 40      88          JP   Z,ERROR1
3A6E FE 0D         89          CP   $0D
3A70 CA 22 40      90          JP   Z,ERROR1
3A73 13            91          INC  DE
3A74 18 ED         92          JR   MID1
3A76               93 MID2
3A76 D9            94          EXX
```

```
3A77 C5            95          PUSH BC
3A78 D9            96          EXX
3A79 C1            97          POP  BC
3A7A               98 MID3
3A7A 1A            99          LD   A,(DE)
3A7B 77           100          LD   (HL),A
3A7C FE 22        101          CP   '"'
3A7E 28 0B        102          JR   Z,MID4
3A80 FE 0D        103          CP   $0D
3A82 28 07        104          JR   Z,MID4
3A84 23           105          INC  HL
3A85 13           106          INC  DE
3A86 0B           107          DEC  BC
3A87 78           108          LD   A,B
3A88 B1           109          OR   C
3A89 20 EF        110          JR   NZ,MID3
3A8B              111 MID4
3A8B 36 0D        112          LD   (HL),$0D
3A8D C3 06 34     113          JP   MAIN
3A90              114 @STRCAT
3A90 D1           115          POP  DE
3A91 E1           116          POP  HL
3A92              117 STRCAT1
3A92 7E           118          LD   A,(HL)
3A93 FE 0D        119          CP   $0D
3A95 CA 0C 3A     120          JP   Z,STRCPY1
3A98 FE 22        121          CP   '"'
3A9A CA 0C 3A     122          JP   Z,STRCPY1
3A9D 23           123          INC  HL
3A9E 18 F2        124          JR   STRCAT1
3AA0              125 @STRLEN
3AA0 E1           126          POP  HL
3AA1 01 00 00     127          LD   BC,0
3AA4              128 STRLEN1
3AA4 7E           129          LD   A,(HL)
3AA5 FE 0D        130          CP   $0D
3AA7 28 08        131          JR   Z,STRLEN2
3AA9 FE 22        132          CP   '"'
3AAB 28 04        133          JR   Z,STRLEN2
3AAD 23           134          INC  HL
3AAE 03           135          INC  BC
3AAF 18 F3        136          JR   STRLEN1
3AB1              137 STRLEN2
3AB1 C5           138          PUSH BC
3AB2 C3 06 34     139          JP   MAIN
3AB5              140 @INSTR
3AB5 C1           141          POP  BC
3AB6 E1           142          POP  HL
3AB7 11 01 00     143          LD   DE,1
3ABA              144 INSTR1
3ABA 7E           145          LD   A,(HL)
3ABB B9           146          CP   C
3ABC 28 0F        147          JR   Z,INSTR3
3ABE FE 0D        148          CP   $0D
3AC0 28 08        149          JR   Z,INSTR2
3AC2 FE 22        150          CP   '"'
3AC4 28 04        151          JR   Z,INSTR2
3AC6 13           152          INC  DE
3AC7 23           153          INC  HL
3AC8 18 F0        154          JR   INSTR1
3ACA              155 INSTR2
3ACA 11 00 00     156          LD   DE,0
3ACD              157 INSTR3
3ACD D5           158          PUSH DE
3ACE C3 06 34     159          JP   MAIN
3AD1              160 @STRCMP
3AD1 E1           161          POP  HL
3AD2 D1           162          POP  DE
3AD3              163 STRCMP1
3AD3 1A           164          LD   A,(DE)
3AD4 FE 0D        165          CP   $0D
3AD6 28 0C        166          JR   Z,STRCMP2
3AD8 FE 22        167          CP   '"'
3ADA 28 08        168          JR   Z,STRCMP2
3ADC 46           169          LD   B,(HL)
3ADD 90           170          SUB  B
3ADE 20 13        171          JR   NZ,STRCMP3
3AE0 23           172          INC  HL
3AE1 13           173          INC  DE
3AE2 18 EF        174          JR   STRCMP1
3AE4              175 STRCMP2
3AE4 01 00 00     176          LD   BC,0
3AE7 7E           177          LD   A,(HL)
3AE8 FE 0D        178          CP   $0D
3AEA 28 0F        179          JR   Z,STRCMP4
3AEC FE 22        180          CP   '"'
3AEE 28 0B        181          JR   Z,STRCMP4
3AF0 0B           182          DEC  BC
3AF1 18 08        183          JR   STRCMP4
3AF3              184 STRCMP3
3AF3 01 01 00     185          LD   BC,1
3AF6 30 03        186          JR   NC,STRCMP4
3AF8 01 FF FF     187          LD   BC,-1
3AFB              188 STRCMP4
3AFB C5           189          PUSH BC
3AFC C3 06 34     190          JP   MAIN
3AFF              191 ;
3AFF              192 ; 32 Bit ｴﾝｻﾞﾝ
3AFF              193 ;
3AFF              194 @LTASU
3AFF C1           195          POP  BC
3B00 D1           196          POP  DE
3B01 E1           197          POP  HL
3B02 09           198          ADD  HL,BC
3B03 EB           199          EX   DE,HL
3B04 C1           200          POP  BC
3B05 ED 4A        201          ADC  HL,BC
3B07 E5           202          PUSH HL
3B08 D5           203          PUSH DE
3B09 C3 06 34     204          JP   MAIN
3B0C              205 @LHIKU
3B0C D1           206          POP  DE
3B0D C1           207          POP  BC
3B0E E1           208          POP  HL
3B0F B7           209          OR   A
3B10 ED 52        210          SBC  HL,DE
3B12 EB           211          EX   DE,HL
3B13 E1           212          POP  HL
3B14 ED 42        213          SBC  HL,BC
3B16 E5           214          PUSH HL
3B17 D5           215          PUSH DE
3B18 C3 06 34     216          JP   MAIN
3B1B              217 @LMLT
3B1B D1           218          POP  DE
3B1C E1           219          POP  HL
3B1D D9           220          EXX
3B1E FD 21 00     221          LD   IY,0
3B21 00
3B22 21 00 00     222          LD   HL,0
3B25 D1           223          POP  DE
3B26 C1           224          POP  BC
3B27 D9           225          EXX
3B28 06 20        226          LD   B,32
3B2A              227 LMLT1
3B2A CB 3C        228          SRL  H
3B2C CB 1D        229          RR   L
3B2E CB 1A        230          RR   D
3B30 CB 1B        231          RR   E
3B32 D9           232          EXX
3B33 30 04        233          JR   NC,LMLT2
3B35 FD 19        234          ADD  IY,DE
3B37 ED 4A        235          ADC  HL,BC
3B39              236 LMLT2
3B39 CB 23        237          SLA  E
3B3B CB 12        238          RL   D
3B3D CB 11        239          RL   C
3B3F CB 10        240          RL   B
3B41 D9           241          EXX
3B42 10 E6        242          DJNZ LMLT1
3B44 D9           243          EXX
3B45 E5           244          PUSH HL
3B46 FD E5        245          PUSH IY
3B48 C3 06 34     246          JP   MAIN
3B4B              247 @LDIV
3B4B CD 6D 3B     248          CALL LDIV
3B4E D9           249          EXX
3B4F C5           250          PUSH BC
3B50 D9           251          EXX
3B51 C5           252          PUSH BC
3B52 C3 06 34     253          JP   MAIN
3B55              254 @LMOD
3B55 CD 6D 3B     255          CALL LDIV
```

```
3B58 D9           256          EXX
3B59 E5           257          PUSH HL
3B5A D9           258          EXX
3B5B E5           259          PUSH HL
3B5C C3 06 34     260          JP   MAIN
3B5F              261 @LDIVMD
3B5F CD 6D 3B     262          CALL LDIV
3B62 D9           263          EXX
3B63 E5           264          PUSH HL
3B64 D9           265          EXX
3B65 E5           266          PUSH HL
3B66 D9           267          EXX
3B67 C5           268          PUSH BC
3B68 D9           269          EXX
3B69 C5           270          PUSH BC
3B6A C3 06 34     271          JP   MAIN
3B6D              272 ;
3B6D              273 ; BC'BC=BC'BC/DE'DE
3B6D              274 ; HL'HL=BC'BC MOD DE'DE
3B6D              275 ;
3B6D              276 LDIV
3B6D FD E1        277          POP  IY
3B6F D1           278          POP  DE
3B70 21 00 00     279          LD   HL,0
3B73 D9           280          EXX
3B74 D1           281          POP  DE
3B75 21 00 00     282          LD   HL,0
3B78 D9           283          EXX
3B79 C1           284          POP  BC
3B7A D9           285          EXX
3B7B C1           286          POP  BC
3B7C D9           287          EXX
3B7D              288 ;
3B7D 3E 20        289          LD   A,32
3B7F              290 LDIV1
3B7F F5           291          PUSH AF
3B80 CB 21        292          SLA  C
3B82 CB 10        293          RL   B
3B84 D9           294          EXX
3B85 CB 11        295          RL   C
3B87 CB 10        296          RL   B
3B89 D9           297          EXX
3B8A ED 6A        298          ADC  HL,HL
3B8C D9           299          EXX
3B8D ED 6A        300          ADC  HL,HL
3B8F D9           301          EXX
3B90 B7           302          OR   A
3B91 ED 52        303          SBC  HL,DE
3B93 D9           304          EXX
3B94 ED 52        305          SBC  HL,DE
3B96 D9           306          EXX
3B97 38 0E        307          JR   C,LDIV2
3B99              308 ;
3B99 03           309          INC  BC
3B9A 78           310          LD   A,B
3B9B B1           311          OR   C
3B9C 20 03        312          JR   NZ,LDIV3
3B9E D9           313          EXX
3B9F 03           314          INC  BC
3BA0 D9           315          EXX
3BA1              316 LDIV3
3BA1 F1           317          POP  AF
3BA2 3D           318          DEC  A
3BA3 20 DA        319          JR   NZ,LDIV1
3BA5 FD E9        320          JP   (IY)
3BA7              321 LDIV2
3BA7 19           322          ADD  HL,DE
3BA8 D9           323          EXX
3BA9 ED 5A        324          ADC  HL,DE
3BAB D9           325          EXX
3BAC 18 F3        326          JR   LDIV3
3BAE              327 @DDIVMOD
3BAE C1           328          POP  BC
3BAF D1           329          POP  DE
3BB0 E1           330          POP  HL
3BB1 CD BC 3B     331          CALL QUOT
3BB4 D9           332          EXX
3BB5 D5           333          PUSH DE
3BB6 D9           334          EXX
3BB7 E5           335          PUSH HL
3BB8 D5           336          PUSH DE
3BB9 C3 06 34     337          JP   MAIN
3BBC              338 ;
3BBC              339 ; HLDE=HLDE/BC
3BBC              340 ;
3BBC              341 ; DE'=HLDE MOD BC
3BBC              342 ;
3BBC              343 QUOT
3BBC F5           344          PUSH AF
3BBD C5           345          PUSH BC
3BBE D9           346          EXX
3BBF C1           347          POP  BC
3BC0 21 00 00     348          LD   HL,0
3BC3 11 00 00     349          LD   DE,0
3BC6 D9           350          EXX
3BC7 3E 20        351          LD   A,32
3BC9              352 QUOT1
3BC9 EB           353          EX   DE,HL
3BCA 29           354          ADD  HL,HL
3BCB EB           355          EX   DE,HL
3BCC ED 6A        356          ADC  HL,HL
3BCE D9           357          EXX
3BCF EB           358          EX   DE,HL
3BD0 ED 6A        359          ADC  HL,HL
3BD2 EB           360          EX   DE,HL
3BD3 ED 6A        361          ADC  HL,HL
3BD5 D5           362          PUSH DE
3BD6 E5           363          PUSH HL
3BD7 CD EA 3B     364          CALL QUOTSB
3BDA E1           365          POP  HL
3BDB D1           366          POP  DE
3BDC 38 03        367          JR   C,QUOT2
3BDE CD EA 3B     368          CALL QUOTSB
3BE1              369 QUOT2
3BE1 D9           370          EXX
3BE2 38 01        371          JR   C,QUOT3
3BE4 1C           372          INC  E
3BE5              373 QUOT3
3BE5 3D           374          DEC  A
3BE6 20 E1        375          JR   NZ,QUOT1
3BE8 F1           376          POP  AF
3BE9 C9           377          RET
3BEA              378 QUOTSB
3BEA EB           379          EX   DE,HL
3BEB B7           380          OR   A
3BEC ED 42        381          SBC  HL,BC
3BEE EB           382          EX   DE,HL
3BEF D0           383          RET  NC
3BF0 67           384          LD   H,A
3BF1 7D           385          LD   A,L
3BF2 D6 01        386          SUB  1
3BF4 6F           387          LD   L,A
3BF5 7C           388          LD   A,H
3BF6 26 00        389          LD   H,0
3BF8 C9           390          RET
3BF9              391 @DMLT
3BF9 C1           392          POP  BC
3BFA D1           393          POP  DE
3BFB 21 00 00     394          LD   HL,0
3BFE D9           395          EXX
3BFF 11 00 00     396          LD   DE,0
3C02 21 00 00     397          LD   HL,0
3C05 D9           398          EXX
3C06 3E 10        399          LD   A,16
3C08              400 DMLT1
3C08 CB 38        401          SRL  B
3C0A CB 19        402          RR   C
3C0C 30 05        403          JR   NC,DMLT2
3C0E 19           404          ADD  HL,DE
3C0F D9           405          EXX
3C10 ED 5A        406          ADC  HL,DE
3C12 D9           407          EXX
3C13              408 DMLT2
3C13 CB 23        409          SLA  E
3C15 CB 12        410          RL   D
3C17 D9           411          EXX
3C18 CB 13        412          RL   E
3C1A CB 12        413          RL   D
3C1C D9           414          EXX
3C1D 3D           415          DEC  A
3C1E 20 E8        416          JR   NZ,DMLT1
3C20 D9           417          EXX
```


▶ヒャッホー，就職できたー。これで「ごくつぶし」じゃなくなる。
寺村 公一 (23) X68000 滋賀県

```
3C21 E5          418              PUSH HL
3C22 D9          419              EXX
3C23 E5          420              PUSH HL
3C24 C3 06 34    421              JP   MAIN
3C27             422 @MLT!
3C27 C1          423              POP  BC
3C28 D1          424              POP  DE
3C29 21 00 00    425              LD   HL,0
3C2C 79          426              LD   A,C
3C2D             427 MLT!1
3C2D B7          428              OR   A
3C2E 28 0C       429              JR   Z,MLT!3
3C30 CB 3F       430              SRL  A
3C32 30 01       431              JR   NC,MLT!2
3C34 19          432              ADD  HL,DE
3C35             433 MLT!2
3C35 CB 23       434              SLA  E
3C37 CB 12       435              RL   D
3C39 C3 2D 3C    436              JP   MLT!1
3C3C             437 MLT!3
3C3C E5          438              PUSH HL
3C3D C3 06 34    439              JP   MAIN
3C40             440 @CMP2
3C40 C1          441              POP  BC
3C41 D1          442              POP  DE
3C42 E1          443              POP  HL
3C43 B7          444              OR   A
3C44 ED 42       445              SBC  HL,BC
3C46 4D          446              LD   C,L
3C47 44          447              LD   B,H
3C48 E1          448              POP  HL
3C49 ED 52       449              SBC  HL,DE
3C4B 38 0A       450              JR   C,CMP2_1
3C4D 7C          451              LD   A,H
3C4E B5          452              OR   L
3C4F B0          453              OR   B
3C50 B1          454              OR   C
3C51 20 09       455              JR   NZ,CMP2_2
3C53             456 CMP2_3
3C53 E5          457              PUSH HL
3C54 C3 06 34    458              JP   MAIN
3C57             459 CMP2_1
3C57 21 FF FF    460              LD   HL,-1
3C5A 18 F7       461              JR   CMP2_3
3C5C             462 CMP2_2
3C5C 21 01 00    463              LD   HL,1
3C5F 18 F2       464              JR   CMP2_3
3C61             465 @CTL
3C61 E1          466              POP  HL
3C62 11 00 00    467              LD   DE,0
3C65 CB 7C       468              BIT  7,H
3C67 28 01       469              JR   Z,CTL1
3C69 1B          470              DEC  DE
3C6A             471 CTL1
3C6A D5          472              PUSH DE
3C6B E5          473              PUSH HL
3C6C C3 06 34    474              JP   MAIN
3C6F             475 @ASCII
3C6F E1          476              POP  HL
3C70 11 00 00    477              LD   DE,0
3C73             478 ASCII1
3C73 7E          479              LD   A,(HL)
3C74 23          480              INC  HL
3C75 FE 0D       481              CP   $0D
3C77 28 08       482              JR   Z,ASCII2
3C79 FE 22       483              CP   '"'
3C7B 28 04       484              JR   Z,ASCII2
3C7D 53          485              LD   D,E
3C7E 5F          486              LD   E,A
3C7F 18 F2       487              JR   ASCII1
3C81             488 ASCII2
3C81 D5          489              PUSH DE
3C82 C3 06 34    490              JP   MAIN
3C85             491 @F<
3C85 D1          492              POP  DE
3C86 E1          493              POP  HL
3C87 01 01 00    494              LD   BC,1
3C8A CB 7A       495              BIT  7,D
3C8C 20 0E       496              JR   NZ,F<1
3C8E CB 7C       497              BIT  7,H
3C90 20 06       498              JR   NZ,F<END
3C92             499 F<4
3C92 B7          500              OR   A
3C93 ED 52       501              SBC  HL,DE
3C95 38 01       502              JR   C,F<END
3C97             503 F<3
3C97 0B          504              DEC  BC
3C98             505 F<END
3C98 C5          506              PUSH BC
3C99 C3 06 34    507              JP   MAIN
3C9C             508 F<1
3C9C CB 7C       509              BIT  7,H
3C9E 28 F7       510              JR   Z,F<3
3CA0             511 F<2
3CA0 18 F0       512              JR   F<4
3CA2             513 @=0
3CA2 E1          514              POP  HL
3CA3 11 01 00    515              LD   DE,1
3CA6 7C          516              LD   A,H
3CA7 B5          517              OR   L
3CA8 28 01       518              JR   Z,@=01
3CAA 1B          519              DEC  DE
3CAB             520 @=01
3CAB D5          521              PUSH DE
3CAC C3 06 34    522              JP   MAIN
3CAF             523 @INC#
3CAF E1          524              POP  HL
3CB0 23          525              INC  HL
3CB1 E5          526              PUSH HL
3CB2 C3 06 34    527              JP   MAIN
3CB5             528 @DEC#
3CB5 E1          529              POP  HL
3CB6 2B          530              DEC  HL
3CB7 E5          531              PUSH HL
3CB8 C3 06 34    532              JP   MAIN
3CBB             533 @PRINT1
3CBB D1          534              POP  DE
3CBC 21 00 00    535              LD   HL.0
3CBF             536 @PRINT11
3CBF CD 2B 3F    537              CALL CVHLDE
3CC2 11 4F 3F    538              LD   DE,@CVBUF
3CC5 CD E8 1F    539              CALL #MSG
3CC8 C3 06 34    540              JP   MAIN
3CCB             541 @PRINT2
3CCB D1          542              POP  DE
3CCC E1          543              POP  HL
3CCD 18 F0       544              JR   @PRINT11
3CCF             545 @PRF
3CCF E1          546              POP  HL
3CD0 11 00 00    547              LD   DE,0
3CD3 CB 7C       548              BIT  7,H
3CD5 28 08       549              JR   Z,PRINTF1
3CD7 3E 2D       550              LD   A,"-"
3CD9 CD F4 1F    551              CALL #PRINT
3CDC CD EC 39    552              CALL NEGATE
3CDF             553 PRINTF1
3CDF EB          554              EX   DE,HL
3CE0 18 DD       555              JR   @PRINT11
3CE2             556 @PRF2
3CE2 D1          557              POP  DE
3CE3 E1          558              POP  HL
3CE4 CB 7C       559              BIT  7,H
3CE6 28 D7       560              JR   Z,@PRINT11
3CE8 3E 2D       561              LD   A,"-"
3CEA CD F4 1F    562              CALL #PRINT
3CED CD F4 39    563              CALL NEGATE2
3CF0 C3 BF 3C    564              JP   @PRINT11
3CF3             565 @STRW
3CF3 E1          566              POP  HL
3CF4 D1          567              POP  DE
3CF5 E5          568              PUSH HL
3CF6 21 00 00    569              LD   HL,0
3CF9             570 STRW1
3CF9 CD 2B 3F    571              CALL CVHLDE
3CFC E1          572              POP  HL
3CFD 11 4F 3F    573              LD   DE,@CVBUF
3D00 C3 0C 3A    574              JP   STRCPY1
3D03             575 @STRL
3D03 C1          576              POP  BC
3D04 D1          577              POP  DE
3D05 E1          578              POP  HL
3D06 C5          579              PUSH BC
3D07 18 F0       580              JR   STRW1
3D09             581 @HEXL
3D09 D1          582              POP  DE
3D0A E1          583              POP  HL
3D0B CD BE 1F    584              CALL #PRTHL
3D0E EB          585              EX   DE,HL
3D0F CD BE 1F    586              CALL #PRTHL
3D12 C3 06 34    587              JP   MAIN
3D15             588 @VAL1
3D15 E1          589              POP  HL
3D16 DD E5       590              PUSH IX
3D18 E5          591              PUSH HL      ;  IX = HL
3D19 DD E1       592              POP  IX      ;
3D1B DD 7E 00    593              LD   A,(IX)
3D1E FE 2D       594              CP   "-"
3D20 28 09       595              JR   Z,@VAL1_1
3D22 CD 6B 3F    596              CALL DECI
3D25             597 @VAL1_2
3D25 DD E1       598              POP  IX
3D27 E5          599              PUSH HL
3D28 C3 06 34    600              JP   MAIN
3D2B             601 @VAL1_1
3D2B DD 23       602              INC  IX
3D2D CD 6B 3F    603              CALL DECI
3D30 CD EC 39    604              CALL NEGATE
3D33 18 F0       605              JR   @VAL1_2
3D35             606 @VAL2
3D35 E1          607              POP  HL
3D36 DD E5       608              PUSH IX
3D38 E5          609              PUSH HL      ;  IX = HL
3D39 DD E1       610              POP  IX      ;
3D3B CD 89 3F    611              CALL HLDEDECI
3D3E DD E1       612              POP  IX
3D40 E5          613              PUSH HL
3D41 D5          614              PUSH DE
3D42 C3 06 34    615              JP   MAIN
3D45             616 @VAL$
3D45 E1          617              POP  HL
3D46 DD E5       618              PUSH IX
3D48 E5          619              PUSH HL      ; IX = HL
3D49 DD E1       620              POP  IX      ;
3D4B CD 85 35    621              CALL #HEX
3D4E DD E1       622              POP  IX
3D50 E5          623              PUSH HL
3D51 C3 06 34    624              JP   MAIN
3D54             625 @INP$
3D54 CD 18 20    626              CALL #CSR
3D57 26 00       627              LD   H,0
3D59 ED 5B 76    628              LD   DE,(#KBFAD)
3D5C 1F
3D5D CD D3 1F    629              CALL #GETL
3D60 1A          630              LD   A,(DE)
3D61 FE 1B       631              CP   $1B
3D63 28 0C       632              JR   Z,INP1
3D65 19          633              ADD  HL,DE
3D66 EB          634              EX   DE,HL
3D67 E1          635              POP  HL
3D68             636 INP2
3D68 1A          637              LD   A,(DE)
3D69 B7          638              OR   A
3D6A 28 06       639              JR   Z,INP3
3D6C 77          640              LD   (HL),A
3D6D 13          641              INC  DE
3D6E 23          642              INC  HL
3D6F 18 F7       643              JR   INP2
3D71             644 INP1
3D71 E1          645              POP  HL
3D72             646 INP3
3D72 36 0D       647              LD   (HL),$0D
3D74 C3 06 34    648              JP   MAIN
3D77             649 @TRANS1
3D77 C1          650              POP  BC
3D78 D1          651              POP  DE
3D79 E1          652              POP  HL
3D7A ED B0       653              LDIR
3D7C C3 06 34    654              JP   MAIN
3D7F             655 @TRANS2
3D7F C1          656              POP  BC
3D80 D1          657              POP  DE
3D81 E1          658              POP  HL
3D82 ED B8       659              LDDR
3D84 C3 06 34    660              JP   MAIN
3D87             661 @FILL
3D87 D1          662              POP  DE
3D88 C1          663              POP  BC
3D89 E1          664              POP  HL
3D8A 0B          665              DEC  BC
3D8B 73          666              LD   (HL),E
3D8C 54          667              LD   D,H
3D8D 5D          668              LD   E,L
3D8E 13          669              INC  DE
3D8F ED B0       670              LDIR
3D91 C3 06 34    671              JP   MAIN
3D94             672 @COPYL
3D94 D1          673              POP  DE
3D95 E1          674              POP  HL
3D96 E5          675              PUSH HL
3D97 D5          676              PUSH DE
3D98 E5          677              PUSH HL
3D99 D5          678              PUSH DE
3D9A C3 06 34    679              JP   MAIN
3D9D             680 @DROPL
3D9D E1          681              POP  HL
3D9E E1          682              POP  HL
3D9F C3 06 34    683              JP   MAIN
3DA2             684 @SWAPD
3DA2 D1          685              POP  DE
3DA3 E1          686              POP  HL
3DA4 D9          687              EXX
3DA5 D1          688              POP  DE
3DA6 E1          689              POP  HL
3DA7 D9          690              EXX
3DA8 E5          691              PUSH HL
3DA9 D5          692              PUSH DE
3DAA D9          693              EXX
3DAB E5          694              PUSH HL
3DAC D5          695              PUSH DE
3DAD C3 06 34    696              JP   MAIN
3DB0             697 @BREAK
3DB0 CD CD 1F    698              CALL #BREAK
3DB3 C2 06 34    699              JP   NZ,MAIN
3DB6 C3 07 41    700              JP   END
3DB9             701 ;
3DB9             702 ; GRAPHIC
3DB9             703 ;
3DB9             704 MAGIC     EQU  $B004
3DB9             705 MAINIT    EQU  $AF00
3DB9             706 ;
3DB9             707 @INIT
3DB9 DD E5       708              PUSH IX
3DBB CD 00 AF    709              CALL MAINIT
3DBE DD 21 CA    710              LD   IX.INITDATA
3DC1 3D
3DC2             711 INIT1
3DC2 CD 04 B0    712              CALL MAGIC
3DC5 DD E1       713              POP  IX
3DC7 C3 06 34    714              JP   MAIN
3DCA             715 INITDATA
3DCA 06          716              DB   6
3DCB 00 00 00    717              DW   0,0,639,199
3DCE 00 7F 02
3DD1 C7 00
3DD3 0A 00 01    718              DB   $0A,0.1,2,3,4,5,6,7
3DD6 02 03 04
3DD9 05 06 07
3DDC             719 CLSDATA
3DDC 07 02 02    720              DB   7,2,2,9,7,2,1,9,7,2,0,9
3DDF 09 07 02
3DE2 01 09 07
3DE5 02 00 09
3DE8 0F          721              DB   $0F
3DE9             722 @COL
3DE9 D1          723              POP  DE
3DEA 21 FA 3D    724              LD   HL,COLDATA+1
3DED 73          725              LD   (HL),E
3DEE 23          726              INC  HL
3DEF D1          727              POP  DE
3DF0 73          728              LD   (HL),E
3DF1             729 COL1
3DF1 DD E5       730              PUSH IX
3DF3 DD 21 F9    731              LD   IX,COLDATA
3DF6 3D
3DF7 18 C9       732              JR   INIT1
3DF9             733 COLDATA
3DF9 07          734              DB   7
3DFA 02          735              DB   2
3DFB 00          736              DB   0
3DFC 0F          737              DB   $0F
3DFD             738 @CLS
3DFD D1          739              POP  DE
3DFE 7B          740              LD   A,E
3DFF FE 03       741              CP   3
3E01 30 10       742              JR   NC,CLS1
3E03 32 1D 3E    743              LD   (CLSDATA1+2),A
3E06 DD E5       744              PUSH IX
3E08 DD 21 1B    745              LD   IX,CLSDATA1
3E0B 3E
3E0C             746 CLS2
3E0C CD 04 B0    747              CALL MAGIC
3E0F DD E1       748              POP  IX
3E11 18 DE       749              JR   COL1
3E13             750 CLS1
3E13 DD E5       751              PUSH IX
3E15 DD 21 DC    752              LD   IX,CLSDATA
3E18 3D
3E19 18 F1       753              JR   CLS2
3E1B             754 CLSDATA1
3E1B 07          755              DB   7
3E1C 02          756              DB   2
3E1D 00          757              DB   0
3E1E 09          758              DB   9
3E1F 0F          759              DB   $0F
3E20             760 @PALET
3E20 21 F4 3E    761              LD   HL,MAGICBUF+9
3E23 36 0F       762              LD   (HL),$0F
3E25 06 08       763              LD   B,8
3E27             764 PALET1
3E27 2B          765              DEC  HL
3E28 D1          766              POP  DE
3E29 73          767              LD   (HL),E
3E2A 10 FB       768              DJNZ PALET1
3E2C 2B          769              DEC  HL
3E2D 36 0A       770              LD   (HL),$0A
3E2F             771 PALET2
3E2F DD E5       772              PUSH IX
3E31 DD 21 EB    773              LD   IX,MAGICBUF
3E34 3E
3E35 CD 04 B0    774              CALL MAGIC
3E38 DD E1       775              POP  IX
3E3A C3 06 34    776              JP   MAIN
3E3D             777 WDATA
3E3D 58          778              LD   E,B
3E3E CB 23       779              SLA  E
3E40 16 00       780              LD   D,0
3E42 19          781              ADD  HL,DE
3E43 23          782              INC  HL
3E44 36 0F       783              LD   (HL),$0F
3E46             784 WDATA1
3E46 D1          785              POP  DE
3E47 2B          786              DEC  HL
3E48 72          787              LD   (HL),D
3E49 2B          788              DEC  HL
3E4A 73          789              LD   (HL),E
3E4B 10 F9       790              DJNZ WDATA1
3E4D 18 E0       791              JR   PALET2
3E4F             792 @WIND
3E4F 21 EB 3E    793              LD   HL,MAGICBUF
3E52 36 06       794              LD   (HL),6
3E54 06 04       795              LD   B,4
3E56 18 E5       796              JR   WDATA
3E58             797 @LINE
3E58 21 EB 3E    798              LD   HL,MAGICBUF
3E5B 36 00       799              LD   (HL),0
3E5D 23          800              INC  HL
3E5E 36 02       801              LD   (HL),2
3E60 06 04       802              LD   B,4
3E62 18 D9       803              JR   WDATA
3E64             804 @SLINE
3E64 21 EB 3E    805              LD   HL,MAGICBUF
3E67 36 01       806              LD   (HL),1
3E69 06 06       807              LD   B,6
3E6B 18 D0       808              JR   WDATA
3E6D             809 @BOX
3E6D 21 EB 3E    810              LD   HL,MAGICBUF
3E70 36 02       811              LD   (HL),2
3E72 06 04       812              LD   B,4
3E74 18 C7       813              JR   WDATA
3E76             814 @TILE
3E76 21 88 3E    815              LD   HL,TILEBUF+3
3E79 D1          816              POP  DE
3E7A 72          817              LD   (HL),D
3E7B 2B          818              DEC  HL
3E7C 73          819              LD   (HL),E
3E7D D1          820              POP  DE
3E7E 2B          821              DEC  HL
3E7F 72          822              LD   (HL),D
3E80 2B          823              DEC  HL
3E81 73          824              LD   (HL),E
3E82 C3 06 34    825              JP   MAIN
3E85             826 TILEBUF
3E85 FF FF       827              DW   $FFFF
3E87 FF FF       828              DW   $FFFF
3E89             829 @BOXFUL
3E89 21 EB 3E    830              LD   HL,MAGICBUF
3E8C 36 04       831              LD   (HL),4
3E8E 06 04       832              LD   B,4
3E90             833 BOXF1
3E90 23          834              INC  HL
3E91 C5          835              PUSH BC
3E92 11 85 3E    836              LD   DE,TILEBUF
3E95 01 04 00    837              LD   BC,4
3E98 EB          838              EX   DE,HL
3E99 ED B0       839              LDIR
3E9B EB          840              EX   DE,HL
3E9C C1          841              POP  BC
3E9D 2B          842              DEC  HL
3E9E 18 9D       843              JR   WDATA
3EA0             844 @TRIANGLE
3EA0 21 EB 3E    845              LD   HL,MAGICBUF
3EA3 36 03       846              LD   (HL),3
3EA5 06 06       847              LD   B,6
3EA7 18 E7       848              JR   BOXF1
3EA9             849 @CIRCLE
3EA9 21 EB 3E    850              LD   HL,MAGICBUF
3EAC 36 05       851              LD   (HL),5
3EAE 06 03       852              LD   B,3
3EB0 18 DE       853              JR   BOXF1
3EB2             854 @DOT
3EB2 21 EB 3E    855              LD   HL,MAGICBUF
3EB5 36 00       856              LD   (HL),0
3EB7 23          857              INC  HL
3EB8 36 01       858              LD   (HL),1
3EBA 06 02       859              LD   B,2
3EBC C3 3D 3E    860              JP   WDATA
3EBF             861 @MAGIC
3EBF DD E3       862              EX   (SP),IX
3EC1 C3 C2 3D    863              JP   INIT1
3EC4             864 @POINT
3EC4 21 EB 3E    865              LD   HL,MAGICBUF
3EC7 36 08       866              LD   (HL),8
3EC9 C1          867              POP  BC
3ECA D1          868              POP  DE
3ECB 23          869              INC  HL
3ECC 73          870              LD   (HL),E
3ECD 23          871              INC  HL
3ECE 72          872              LD   (HL),D
3ECF 23          873              INC  HL
3ED0 71          874              LD   (HL),C
3ED1 23          875              INC  HL
3ED2 70          876              LD   (HL),B
3ED3 23          877              INC  HL
3ED4 36 0F       878              LD   (HL),$0F
3ED6 DD E5       879              PUSH IX
3ED8 DD 21 EB    880              LD   IX,MAGICBUF
3EDB 3E
3EDC CD 04 B0    881              CALL MAGIC
3EDF DD E1       882              POP  IX
3EE1 3A 02 C2    883              LD   A,($C202)
3EE4 6F          884              LD   L,A
3EE5 26 00       885              LD   H,0
3EE7 E5          886              PUSH HL
3EE8 C3 06 34    887              JP   MAIN
3EEB             888 MAGICBUF
3EEB 00 00 00    889              DS   32
```

▶先日ダンジョンマスターを買いに行くと売り切れだった。なぜかというと，合格発表シーズンのためX68000がよく出ているためダンジョンマスターも一緒に買っていったりしているのだ。自分もこんなふうに買ったのにすっかり忘れていた。次からは電話で確かめてからこようと反省して家に帰った。　　松井　克之(17) X68000ACE-HD　滋賀県

```
3EEE 00 00 00
3EF1 00 00 00
3EF4 00 00 00
3EF7 00 00 00
3EFA 00 00 00
3EFD 00 00 00
3F00 00 00 00
3F03 00 00 00
3F06 00 00 00
3F09 00 00
3F0B             890 ;
3F0B             891 #LPTON   EQU  $1FD9
3F0B             892 #LPTOF   EQU  $1FD6
3F0B             893 #SDVSW   EQU  $2027
3F0B             894 ;
3F0B             895 @PRON
3F0B CD D9 1F    896          CALL #LPTON
3F0E C3 06 34    897          JP   MAIN
3F11             898 @PROFF
3F11 CD D6 1F    899          CALL #LPTOF
3F14 C3 06 34    900          JP   MAIN
3F17             901 @PEEK#
3F17 E1          902          POP  HL
3F18 CD 94 1F    903          CALL #PEEK
3F1B 6F          904          LD   L,A
3F1C 26 00       905          LD   H,0
3F1E E5          906          PUSH HL
3F1F C3 06 34    907          JP   MAIN
3F22             908 @POKE#
3F22 C1          909          POP  BC
3F23 E1          910          POP  HL
3F24 79          911          LD   A,C
3F25 CD 9A 1F    912          CALL #POKE
3F28 C3 06 34    913          JP   MAIN
3F2B             914 ;
3F2B             915 CVHLDE
3F2B 01 00 00    916          LD   BC,0
3F2E             917 CVHLDE1
3F2E C5          918          PUSH BC
3F2F 01 0A 00    919          LD   BC,10
3F32 CD BC 3B    920          CALL QUOT
3F35 D9          921          EXX
3F36 3E 30       922          LD   A,'0'
3F38 83          923          ADD  A,E
3F39 D9          924          EXX
3F3A C1          925          POP  BC
3F3B 03          926          INC  BC
3F3C F5          927          PUSH AF
3F3D 7C          928          LD   A,H
3F3E B5          929          OR   L
3F3F B2          930          OR   D
3F40 B3          931          OR   E
3F41 20 EB       932          JR   NZ,CVHLDE1
3F43 41          933          LD   B,C
3F44 21 4F 3F    934          LD   HL,@CVBUF
3F47             935 CVHLDE2
3F47 F1          936          POP  AF
3F48 77          937          LD   (HL),A
3F49 23          938          INC  HL
3F4A 10 FB       939          DJNZ CVHLDE2
3F4C 36 0D       940          LD   (HL),$0D
3F4E C9          941          RET
3F4F             942 @CVBUF
3F4F 00 00 00    943          DS 12
3F52 00 00 00
3F55 00 00 00
3F58 00 00 00
3F5B             944 ;
3F5B             945 ; SPACE/TAB SKIP
3F5B             946 ;
3F5B             947 SPC_SKIP1
3F5B DD 23       948          INC  IX
3F5D             949 SPC_SKIP
3F5D DD 7E 00    950          LD   A,(IX)
3F60 FE 20       951          CP   ' '
3F62 CA 5B 3F    952          JP   Z,SPC_SKIP1
3F65 FE 09       953          CP   $09          ;TAB CODE
3F67 CA 5B 3F    954          JP   Z,SPC_SKIP1
3F6A C9          955          RET
3F6B             956 ;
3F6B             957 ; CONVERT DECIMAL TO HL
3F6B             958 ;
3F6B             959 DECI
3F6B 21 00 00    960          LD   HL,0
3F6E             961 DECI1
3F6E DD 7E 00    962          LD   A,(IX)
3F71 FE 30       963          CP   '0'
3F73 D8          964          RET  C
3F74 FE 3A       965          CP   '9'+1
3F76 D0          966          RET  NC
3F77 DD 23       967          INC  IX
3F79 29          968          ADD  HL,HL        ;HL=HL * 10
3F7A 54          969          LD   D,H
3F7B 5D          970          LD   E,L
3F7C 29          971          ADD  HL,HL
3F7D 29          972          ADD  HL,HL
3F7E 19          973          ADD  HL,DE
3F7F D6 30       974          SUB  '0'
3F81 85          975          ADD  A,L
3F82 30 01       976          JR   NC,DECI2
3F84 24          977          INC  H
3F85             978 DECI2
3F85 6F          979          LD   L,A
3F86 C3 6E 3F    980          JP   DECI1
3F89             981 HLDEDECI
3F89 21 00 00    982          LD   HL,0
3F8C 11 00 00    983          LD   DE,0
3F8F             984 HLDE_1
3F8F DD 7E 00    985          LD   A,(IX)
3F92 FE 30       986          CP   '0'
3F94 D8          987          RET  C
3F95 FE 3A       988          CP   '9'+1
3F97 D0          989          RET  NC
3F98 DD 23       990          INC  IX
3F9A D6 30       991          SUB  '0'
3F9C EB          992          EX   DE,HL     ;HLDE *2
3F9D 29          993          ADD  HL,HL
3F9E EB          994          EX   DE,HL
3F9F ED 6A       995          ADC  HL,HL
3FA1             996 ;
3FA1 44          997          LD   B,H
3FA2 4D          998          LD   C,L
3FA3 C5          999          PUSH BC
3FA4 42         1000          LD   B,D
3FA5 4B         1001          LD   C,E
3FA6            1002 ;
3FA6 EB         1003          EX   DE,HL
3FA7 29         1004          ADD  HL,HL
3FA8 EB         1005          EX   DE,HL
3FA9 ED 6A      1006          ADC  HL,HL
3FAB EB         1007          EX   DE,HL
3FAC 29         1008          ADD  HL,HL
3FAD EB         1009          EX   DE,HL
3FAE ED 6A      1010          ADC  HL,HL
3FB0            1011 ;
3FB0 EB         1012          EX   DE,HL
3FB1 09         1013          ADD  HL,BC
3FB2 EB         1014          EX   DE,HL
3FB3 C1         1015          POP  BC
3FB4 ED 4A      1016          ADC  HL,BC
3FB6            1017 ;
3FB6 06 00      1018          LD   B,0
3FB8 4F         1019          LD   C,A
3FB9 EB         1020          EX   DE,HL
3FBA 09         1021          ADD  HL,BC
3FBB EB         1022          EX   DE,HL
3FBC 0E 00      1023          LD   C,0
3FBE ED 4A      1024          ADC  HL,BC
3FC0 18 CD      1025          JR   HLDE_1
3FC2            1026 ;
3FC2            1027 ; CONVERT HL TO DECIMAL
3FC2            1028 ;
3FC2            1029 CVHLD
3FC2 DD E5      1030          PUSH IX
3FC4 DD 21 FF   1031          LD   IX,DTBL
3FC7 3F
3FC8 FD 21 09   1032          LD   IY,CVTBL
3FCB 40
3FCC 3E 04      1033          LD   A,4
3FCE 06 05      1034          LD   B,5
3FD0            1035 CVHLD1
3FD0 0E 2F      1036          LD   C,'0'-1
3FD2 DD 5E 00   1037          LD   E,(IX)
3FD5 DD 56 01   1038          LD   D,(IX+1)
3FD8            1039 CVHLD2
3FD8 0C         1040          INC  C
3FD9 B7         1041          OR   A
3FDA ED 52      1042          SBC  HL,DE
3FDC 30 FA      1043          JR   NC,CVHLD2
3FDE 19         1044          ADD  HL,DE
3FDF B7         1045          OR   A
3FE0 C4 F1 3F   1046          CALL NZ,CVHLD3
3FE3 FD 71 00   1047          LD   (IY),C
3FE6 DD 23      1048          INC  IX
3FE8 DD 23      1049          INC  IX
3FEA FD 23      1050          INC  IY
3FEC 10 E2      1051          DJNZ CVHLD1
3FEE DD E1      1052          POP  IX
3FF0 C9         1053          RET
3FF1            1054 CVHLD3
3FF1 3D         1055          DEC  A
3FF2 08         1056          EX   AF,AF'
3FF3 3E 30      1057          LD   A,'0'
3FF5 B9         1058          CP   C
3FF6 20 04      1059          JR   NZ,CVHLD4
3FF8 08         1060          EX   AF,AF'
3FF9 0E 20      1061          LD   C,' '
3FFB C9         1062          RET
3FFC            1063 CVHLD4
3FFC 08         1064          EX   AF,AF'
3FFD AF         1065          XOR  A
3FFE C9         1066          RET
3FFF            1067 DTBL
3FFF 10 27 E8   1068          DW   10000,1000,100,10,1
4002 03 64 00
4005 0A 00 01
4008 00
4009            1069 CVTBL
4009 00 00 00   1070          DS   5
400C 00 00
400E            1071 @DEC2
400E CD C2 3F   1072          CALL CVHLD
4011 FD 21 09   1073          LD   IY,CVTBL
4014 40
4015 06 05      1074          LD   B,5
4017            1075 @DECI1
4017 FD 7E 00   1076          LD   A,(IY)
401A CD F4 1F   1077          CALL #PRINT
401D FD 23      1078          INC  IY
401F 10 F6      1079          DJNZ @DECI1
4021 C9         1080          RET
4022            1081 ERROR1
4022 11 58 40   1082          LD   DE,ERR1
4025 C3 AA 40   1083          JP   ERROR
4028            1084 ERROR2
4028 11 65 40   1085          LD   DE,ERR2
402B C3 AA 40   1086          JP   ERROR
402E            1087 ERROR3
402E 11 74 40   1088          LD   DE,ERR3
4031 C3 AA 40   1089          JP   ERROR
4034            1090 ERROR4
4034 ED 7B CB   1091          LD   SP,(CPUSTK)
4037 46
4038 CD EB 1F   1092          CALL #NL
403B 2A CD 46   1093          LD   HL,(LINE_WR)
403E CD 0E 40   1094          CALL @DEC2
4041 3E 2D      1095          LD   A,'-'
4043 CD F4 1F   1096          CALL #PRINT
4046 11 87 40   1097          LD   DE,ERR4
4049 C3 B1 40   1098          JP   ERROR'
404C            1099 ERROR5
404C 11 97 40   1100          LD   DE,ERR5
404F C3 AA 40   1101          JP   ERROR
4052            1102 STOP
4052 11 A4 40   1103          LD   DE,@STOP
4055 C3 AA 40   1104          JP   ERROR
4058            1105 ERR1
4058 53 59 4E   1106          DM   "SYNTAX ERROR"
405B 54 41 58
405E 20 45 52
4061 52 4F 52
4064 00         1107          DB   0
4065            1108 ERR2
4065 53 54 41   1109          DM   "STACK EMPTY   "
4068 43 4B 20
406B 45 4D 50
406E 54 59 20
4071 20 20
4073 00         1110          DB   0
4074            1111 ERR3
4074 52 45 54   1112          DM   "RETURN STACK EMPTY"
4077 55 52 4E
407A 20 53 54
407D 41 43 4B
4080 20 45 4D
4083 50 54 59
4086 00         1113          DB   0
4087            1114 ERR4
4087 55 4E 44   1115          DM   "UNDEFINED LABEL"
408A 45 46 49
408D 4E 45 44
4090 20 4C 41
4093 42 45 4C
4096 00         1116          DB   0
4097            1117 ERR5
4097 4F 55 54   1118          DM   "OUT OF LABEL"
409A 20 4F 46
409D 20 4C 41
40A0 42 45 4C
40A3 00         1119          DB   0
40A4            1120 @STOP
40A4 42 52 45   1121          DM   "BREAK"
40A7 41 4B
40A9 00         1122          DB   0
40AA            1123 ERROR
40AA ED 7B CB   1124          LD   SP,(CPUSTK)
40AD 46
40AE CD EB 1F   1125          CALL #NL
40B1            1126 ERROR'
40B1 CD E5 1F   1127          CALL #MSX
40B4 CD EB 1F   1128          CALL #NL
40B7            1129 ;
40B7 ED 5B BF   1130          LD   DE,(TEXTS)
40BA 33
40BB DD E5      1131          PUSH IX
40BD E1         1132          POP  HL
40BE AF         1133          XOR  A
40BF ED 52      1134          SBC  HL,DE
40C1 DA 07 41   1135          JP   C,END
40C4            1136 ;
40C4 21 01 00   1137          LD   HL,1
40C7 22 CD 46   1138          LD   (LINE_WR),HL
40CA DD E5      1139          PUSH IX
40CC E1         1140          POP  HL
40CD 2B         1141          DEC  HL
40CE            1142 @LNSEARCH
40CE ED 53 D1   1143          LD   (LINE_TOP),DE
40D1 46
40D2 AF         1144          XOR  A
40D3 ED 52      1145          SBC  HL,DE
40D5 CA F5 40   1146          JP   Z,@PRTLINE
40D8 19         1147          ADD  HL,DE
40D9            1148 @LSKIP
40D9 1A         1149          LD   A,(DE)
40DA FE 0D      1150          CP   $0D
40DC 28 03      1151          JR   Z,@LSKIP2
40DE 13         1152          INC  DE
40DF 18 F8      1153          JR   @LSKIP
40E1            1154 @LSKIP2
40E1 AF         1155          XOR  A
40E2 ED 52      1156          SBC  HL,DE
40E4 28 0F      1157          JR   Z,@PRTLINE
40E6 38 0D      1158          JR   C,@PRTLINE
40E8 19         1159          ADD  HL,DE
40E9 ED 4B CD   1160          LD   BC,(LINE_WR)
40EC 46
40ED 03         1161          INC  BC
40EE ED 43 CD   1162          LD   (LINE_WR),BC
40F1 46
40F2 13         1163          INC  DE
40F3 18 D9      1164          JR   @LNSEARCH
40F5            1165 @PRTLINE
40F5 2A CD 46   1166          LD   HL,(LINE_WR)
40F8 CD 0E 40   1167          CALL @DEC2
40FB 3E 3A      1168          LD   A,':'
40FD CD F4 1F   1169          CALL #PRINT
4100 ED 5B D1   1170          LD   DE,(LINE_TOP)
4103 46
4104 CD E8 1F   1171          CALL #MSG
4107            1172 END
4107 ED 7B CB   1173          LD   SP,(CPUSTK)
410A 46
410B CD EB 1F   1174          CALL #NL
410E C3 31 30   1175          JP   E_MAIN
4111            1176 ;
4111            1177 ; Semi Compailer
4111            1178 ;
4111            1179 COMP
4111 CD B2 1F   1180          CALL #HLHEX
4114 DA B5 33   1181          JP   C,ERR
4117 E5         1182          PUSH HL
4118 FD E1      1183          POP  IY
411A DD 2A BF   1184          LD   IX,(TEXTS)
411D 33
411E DD E5      1185          PUSH  IX     ;
4120 D1         1186          POP   DE     ;
4121 CD E8 1F   1187          CALL  #MSG   ;
4124 CD EE 1F   1188          CALL  #LTNL  ;
4127 DD 7E 00   1189          LD   A,(IX)  ;
412A B7         1190          OR   A       ;
412B CA A5 42   1191          JP   Z,COMPEND ;
412E            1192 COMP1
412E CD 5D 3F   1193          CALL SPC_SKIP
4131 DD 7E 00   1194          LD   A,(IX)
4134 FD 77 00   1195          LD   (IY),A
4137 DD 23      1196          INC  IX
4139 B7         1197          OR   A
413A CA A5 42   1198          JP   Z,COMPEND
413D            1199 ;
413D FE 27      1200          CP   "'"
413F CA A7 41   1201          JP   Z,_LONGT
4142 FE 24      1202          CP   '$'
4144 CA C0 41   1203          JP   Z,_HEXT
4147 FE 22      1204          CP   '"'
4149 CA EF 41   1205          JP   Z,_STRT
414C FE 23      1206          CP   '#'
414E CA 14 42   1207          JP   Z,_HENW
4151 FE 25      1208          CP   '%'
4153 CA 5F 42   1209          JP   Z,_LABEL
4156 FE 2E      1210          CP   '.'
4158 CA 2F 42   1211          JP   Z,_LET
415B FE 3B      1212          CP   ';'
415D CA C5 41   1213          JP   Z,_REM
4160 FE 2D      1214          CP   "-"         ;
4162 CA 8C 41   1215          JP   Z,MINUS?    ;
4165 FE 0D      1216          CP   $0D
4167 CA D1 41   1217          JP   Z,_CR
416A FE 30      1218          CP   '0'
416C DA 63 42   1219          JP   C,_KAISEKI
416F FE 3A      1220          CP   '9'+1
4171 D2 63 42   1221          JP   NC,_KAISEKI
4174 DD 2B      1222          DEC  IX
4176 CD 6B 3F   1223          CALL DECI
4179            1224 _TEI1
4179 0E 8F      1225          LD   C,143
417B            1226 _TEI2
417B FD 71 00   1227          LD   (IY),C
417E FD 75 01   1228          LD   (IY+1),L
4181 FD 74 02   1229          LD   (IY+2),H
4184            1230 _TEI3
4184 FD 23      1231          INC  IY
4186 FD 23      1232          INC  IY
4188 FD 23      1233          INC  IY
418A 18 A2      1234          JR   COMP1
418C            1235 MINUS?                    ;
418C 47         1236          LD   B,A          ;
418D DD 7E 00   1237          LD   A,(IX)       ;
4190 FE 20      1238          CP   " "          ;
4192 78         1239          LD   A,B          ;
4193 CA 63 42   1240          JP   Z,_KAISEKI   ;
4196 DD 7E 00   1241          LD   A,(IX)       ;
4199 FE 09      1242          CP   09           ;
419B 78         1243          LD   A,B          ;
419C CA 63 42   1244          JP   Z,_KAISEKI   ;
419F CD 6B 3F   1245          CALL DECI         ;
41A2 CD EC 39   1246          CALL NEGATE       ;
41A5 18 D2      1247          JR   _TEI1        ;
41A7            1248 _LONGT
41A7 CD 89 3F   1249          CALL HLDEDECI
41AA FD 36 00   1250          LD   (IY),148
41AD 94
41AE FD 75 01   1251          LD   (IY+1),L
41B1 FD 74 02   1252          LD   (IY+2),H
41B4 FD 73 03   1253          LD   (IY+3),E
41B7 FD 72 04   1254          LD   (IY+4),D
41BA FD 23      1255          INC  IY
41BC FD 23      1256          INC  IY
41BE 18 C4      1257          JR   _TEI3
41C0            1258 _HEXT
41C0 CD 85 35   1259          CALL #HEX
41C3 18 B4      1260          JR   _TEI1
41C5            1261 _REM
41C5 DD 7E 00   1262          LD   A,(IX)
41C8 FE 0D      1263          CP   $0D
41CA CA 2E 41   1264          JP   Z,COMP1
41CD DD 23      1265          INC  IX
41CF 18 F4      1266          JR   _REM
41D1            1267 _CR
41D1 FD 36 00   1268          LD   (IY),255    ;ｸｷﾞﾘ
41D4 FF
41D5 FD 23      1269          INC  IY
41D7 DD 7E 00   1270          LD    A,(IX)   ;
41DA B7         1271          OR    A        ;
41DB CA 2E 41   1272          JP    Z,COMP1  ;
41DE DD E5      1273          PUSH  IX       ;
41E0 D1         1274          POP   DE       ;
41E1 CD E8 1F   1275          CALL  #MSG     ;
41E4 CD EE 1F   1276          CALL  #LTNL    ;
41E7 CD C7 1F   1277          CALL  #PAUSE   ;
41EA D4 42      1278          DW    COMPBRK  ;
41EC C3 2E 41   1279          JP   COMP1
41EF            1280 _STRT
41EF FD 36 00   1281          LD   (IY),138
41F2 8A
41F3 FD 23      1282          INC  IY
41F5            1283 _STRT1
41F5 DD 7E 00   1284          LD   A,(IX)
41F8 FD 77 00   1285          LD   (IY),A
41FB FE 22      1286          CP   '"'
41FD 28 0A      1287          JR   Z,_STRT2
41FF FE 0D      1288          CP   $0D
4201 28 06      1289          JR   Z,_STRT2
4203 DD 23      1290          INC  IX
4205 FD 23      1291          INC  IY
4207 18 EC      1292          JR   _STRT1
4209            1293 _STRT2
4209 FD 36 00   1294          LD   (IY),'"'
420C 22
420D FD 23      1295          INC  IY
420F DD 23      1296          INC  IX
4211 C3 2E 41   1297          JP   COMP1
4214            1298 _HENW
4214 0E 8E      1299          LD   C,142
4216 DD 7E 00   1300          LD   A,(IX)
4219 FE 23      1301          CP   "#"
421B 20 04      1302          JR   NZ,_HEN
421D DD 23      1303          INC  IX
421F 0E 93      1304          LD   C,147      ;_HENL
4221            1305 _HEN
4221 C5         1306          PUSH BC
4222 CD 4B 35   1307          CALL VAR_ADRS
4225 B7         1308          OR   A
4226 11 E6 47   1309          LD   DE,VAR
4229 ED 52      1310          SBC  HL,DE
422B C1         1311          POP  BC
422C C3 7B 41   1312          JP   _TEI2
422F            1313 _LET
422F DD 7E 00   1314          LD   A,(IX)
4232 FE 2E      1315          CP   '.'
4234 28 04      1316          JR   Z,_LET2
4236 0E 8D      1317          LD   C,141
4238 18 E7      1318          JR   _HEN
423A            1319 _LET2
423A DD 23      1320          INC  IX
423C 0E 92      1321          LD   C,146
423E 18 E1      1322          JR   _HEN
```


▶ X68000の新シリーズが，某新聞で発表された。別に買うわけじゃないけどやっぱり気になる。　　今井　彰彦（25）X68000，X1C/turbo II，MZ-80K　大阪府

```
4240            1323 _INC
4240 CD 5D 3F   1324            CALL SPC_SKIP
4243 0E 90      1325            LD   C,144
4245 18 DA      1326            JR   _HEN
4247            1327 _DEC
4247 CD 5D 3F   1328            CALL SPC_SKIP
424A 0E 91      1329            LD   C,145
424C 18 D3      1330            JR   _HEN
424E            1331 _GOTO
424E 0E 8B      1332            LD   C,139
4250            1333 _GOTO1
4250 C5         1334            PUSH BC
4251 CD 5D 3F   1335            CALL SPC_SKIP
4254 CD 6B 3F   1336            CALL DECI
4257 C1         1337            POP  BC
4258 C3 7B 41   1338            JP   _TEI2
425B            1339 _GOSUB
425B 0E 8C      1340            LD   C,140
425D 18 F1      1341            JR   _GOTO1
425F            1342 _LABEL
425F 0E 88      1343            LD   C,136
4261 18 ED      1344            JR   _GOTO1
4263            1345 _KAISEKI
4263 47         1346            LD   B,A
4264 DD 7E 00   1347            LD   A,(IX)
4267 FE 20      1348            CP   ' '
4269 28 0D      1349            JR   Z,_KAISEKI1
426B FE 09      1350            CP   $09
426D 28 09      1351            JR   Z,_KAISEKI1
426F FE 0D      1352            CP   $0D
4271 28 05      1353            JR   Z,_KAISEKI1
4273 DD 23      1354            INC  IX
4275 80         1355            ADD  A,B
4276 18 EB      1356            JR   _KAISEKI
4278            1357 _KAISEKI1
4278 78         1358            LD   A,B
4279 FE 39      1359            CP   $39
427B 28 D1      1360            JR   Z,_GOTO
427D FE 80      1361            CP   $80
427F 28 DA      1362            JR   Z,_GOSUB
4281 FE DA      1363            CP   $DA
4283 28 BB      1364            JR   Z,_INC
4285 FE CC      1365            CP   $CC
4287 28 BE      1366            JR   Z,_DEC
4289 FE 8F      1367            CP   $8F                   ;IF ?
428B 20 04      1368            JR   NZ,_KAISEKI2
428D 3E 89      1369            LD   A,137                 ;@_IF
428F 18 0C      1370            JR   _KAISEKI3
4291            1371 _KAISEKI2
4291 21 19 44   1372            LD   HL,COM_TBL
4294 85         1373            ADD  A,L
4295 30 01      1374            JR   NC,_KAI_SKIP
4297 24         1375            INC  H
4298            1376 _KAI_SKIP
4298 6F         1377            LD   L,A
4299 7E         1378            LD   A,(HL)
429A B7         1379            OR  A
429B 28 26      1380            JR  Z,COMPERR
429D            1381 _KAISEKI3
429D FD 77 00   1382            LD   (IY),A
42A0 FD 23      1383            INC  IY
42A2 C3 2E 41   1384            JP   COMP1
42A5            1385 COMPEND
42A5 CD EB 1F   1386            CALL #NL
42A8 CD E2 1F   1387            CALL #MPRINT
42AB 4F 42 4A   1388            DM   "OBJECT END:"
42AE 45 43 54
42B1 20 45 4E
42B4 44 3A
42B6 00         1389            DB   0
42B7 FD E5      1390            PUSH IY
42B9 E1         1391            POP  HL
42BA CD BE 1F   1392            CALL #PRTHL
42BD CD EB 1F   1393            CALL #NL
42C0 C3 31 30   1394            JP   E_MAIN
42C3            1395 COMPERR
42C3 CD EB 1F   1396            CALL #NL
42C6 CD E2 1F   1397            CALL #MPRINT
42C9 53 59 4E   1398            DM   "SYNTAX?" ;
42CC 54 41 58
42CF 3F
42D0 00         1399            DB   0
42D1 C3 31 30   1400            JP   E_MAIN
42D4            1401 COMPBRK                          ;
42D4 CD EB 1F   1402            CALL  #NL              ;
42D7 CD E2 1F   1403            CALL  #MPRINT          ;
42DA 42 52 45   1404            DM    "BREAK"          ;
42DD 41 4B
42DF 00         1405            DB    00               ;
42E0 C3 31 30   1406            JP   E_MAIN            ;
42E3            1407 ;
42E3            1408 CCMAIN
42E3 DD 7E 00   1409            LD   A,(IX)
42E6 DD 23      1410            INC  IX
42E8 B7         1411            OR   A
42E9 CA 07 41   1412            JP   Z,END
42EC FE FF      1413            CP   255
42EE 28 F3      1414            JR   Z,CCMAIN
42F0 6F         1415            LD   L,A
42F1 26 00      1416            LD   H,0
42F3 29         1417            ADD  HL,HL
42F4 11 19 45   1418            LD   DE,JUMPTBL
42F7 19         1419            ADD  HL,DE
42F8 5E         1420            LD   E,(HL)
42F9 23         1421            INC  HL
42FA 56         1422            LD   D,(HL)
42FB EB         1423            EX   DE,HL
42FC E9         1424            JP   (HL)
42FD            1425 @_WORDT
42FD DD 6E 00   1426            LD   L,(IX)
4300 DD 66 01   1427            LD   H,(IX+1)
4303 E5         1428            PUSH HL
4304            1429 @_WORDT1
4304 DD 23      1430            INC  IX
4306 DD 23      1431            INC  IX
4308 18 D9      1432            JR   CCMAIN
430A            1433 @_LONGT
430A DD 5E 00   1434            LD   E,(IX)
430D DD 56 01   1435            LD   D,(IX+1)
4310 D5         1436            PUSH DE
4311 DD 23      1437            INC  IX
4313 DD 23      1438            INC  IX
4315 18 E6      1439            JR   @_WORDT
4317            1440 @_GETADRS
4317 DD 6E 00   1441            LD   L,(IX)
431A DD 66 01   1442            LD   H,(IX+1)
431D 11 E6 47   1443            LD   DE,VAR
4320 19         1444            ADD  HL,DE
4321 C9         1445            RET
4322            1446 @_HENW
4322 CD 17 43   1447            CALL @_GETADRS
4325            1448 @_HENW1
4325 5E         1449            LD   E,(HL)
4326 23         1450            INC  HL
4327 56         1451            LD   D,(HL)
4328 D5         1452            PUSH DE
4329 18 D9      1453            JR   @_WORDT1
432B            1454 @_HENL
432B CD 17 43   1455            CALL @_GETADRS
432E 4E         1456            LD   C,(HL)
432F 23         1457            INC  HL
4330 46         1458            LD   B,(HL)
4331 23         1459            INC  HL
4332 C5         1460            PUSH BC
4333 18 F0      1461            JR   @_HENW1
4335            1462 @_LETW
4335 CD 17 43   1463            CALL @_GETADRS
4338 D1         1464            POP  DE
4339            1465 @_LETW1
4339 73         1466            LD   (HL),E
433A 23         1467            INC  HL
433B 72         1468            LD   (HL),D
433C 18 C6      1469            JR   @_WORDT1
433E            1470 @_LETL
433E CD 17 43   1471            CALL @_GETADRS
4341 D1         1472            POP  DE
4342 C1         1473            POP  BC
4343 71         1474            LD   (HL),C
4344 23         1475            INC  HL
4345 70         1476            LD   (HL),B
4346 23         1477            INC  HL
4347 18 F0      1478            JR   @_LETW1
4349            1479 @_INC
4349 CD 17 43   1480            CALL @_GETADRS
434C DD 23      1481            INC  IX
434E DD 23      1482            INC  IX
4350 C3 32 37   1483            JP   @INC1
4353            1484 @_DEC
4353 CD 17 43   1485            CALL @_GETADRS
4356 DD 23      1486            INC  IX
4358 DD 23      1487            INC  IX
435A C3 42 37   1488            JP   @DEC1
435D            1489 @_GOTO
435D DD 6E 00   1490            LD   L,(IX)
4360 DD 66 01   1491            LD   H,(IX+1)
4363 C3 80 43   1492            JP   __GOTO
4366            1493 @_GOSUB
4366 DD 6E 00   1494            LD   L,(IX)
4369 DD 66 01   1495            LD   H,(IX+1)
436C DD 23      1496            INC  IX
436E DD 23      1497            INC  IX
4370 C3 F6 43   1498            JP   __GOSUB
4373            1499 @_LABEL
4373 DD 6E 00   1500            LD   L,(IX)
4376 DD 66 01   1501            LD   H,(IX+1)
4379 DD 23      1502            INC  IX
437B DD 23      1503            INC  IX
437D C3 DD 34   1504            JP   LABEL1
4380            1505 __GOTO
4380 29         1506            ADD  HL,HL
4381 CD 03 39   1507            CALL #LABELR
4384 78         1508            LD   A,B
4385 B1         1509            OR   C
4386 CA 8F 43   1510            JP   Z,__GOTO1
4389 C5         1511            PUSH BC
438A DD E1      1512            POP  IX
438C C3 E3 42   1513            JP   CCMAIN
438F            1514 __GOTO1
438F DD 22 CF   1515            LD   (LINE_WR1),IX
4392 46
4393 AF         1516            XOR  A
4394 CB 3C      1517            SRL  H                     ;HL=HL/2
4396 CB 1D      1518            RR   L
4398 22 CD 46   1519            LD   (LINE_WR),HL
439B DD 2A E0   1520            LD   IX,(CCTEXT)
439E 46
439F            1521 __GOTO2
439F DD 7E 00   1522            LD   A,(IX)
43A2 FE 88      1523            CP   136
43A4 C2 C0 43   1524            JP   NZ,_G_SKIP
43A7 DD 23      1525            INC  IX
43A9 DD 6E 00   1526            LD   L,(IX)
43AC DD 66 01   1527            LD   H,(IX+1)
43AF DD 23      1528            INC  IX
43B1 DD 23      1529            INC  IX
43B3 ED 5B CD   1530            LD   DE,(LINE_WR)
43B6 46
43B7 B7         1531            OR   A
43B8 ED 52      1532            SBC  HL,DE
43BA 20 04      1533            JR   NZ,_G_SKIP
43BC 19         1534            ADD  HL,DE
43BD C3 DD 34   1535            JP   LABEL1
43C0            1536 _G_SKIP
43C0 CD C5 43   1537            CALL T_SKIP
43C3 18 DA      1538            JR   __GOTO2
43C5            1539 ;
43C5            1540 T_SKIP
43C5 DD 7E 00   1541            LD   A,(IX)
43C8 DD 23      1542            INC  IX
43CA FE FF      1543            CP   255
43CC C8         1544            RET  Z
43CD            1545 ;
43CD FE 8A      1546            CP   138
43CF 30 02      1547            JR   NC,__SKIP
43D1 18 F2      1548            JR   T_SKIP
43D3            1549 __SKIP
43D3 FE 8A      1550            CP   138
43D5 28 14      1551            JR   Z,_STRSKIP
43D7 FE 92      1552            CP   146
43D9 30 06      1553            JR   NC,__SKIP4
43DB DD 23      1554            INC  IX
43DD DD 23      1555            INC  IX
43DF 18 E4      1556            JR   T_SKIP
43E1            1557 __SKIP4
43E1 DD 23      1558            INC  IX
43E3 DD 23      1559            INC  IX
43E5 DD 23      1560            INC  IX
43E7 DD 23      1561            INC  IX
43E9 18 DA      1562            JR   T_SKIP
43EB            1563 _STRSKIP
43EB DD 7E 00   1564            LD   A,(IX)
43EE DD 23      1565            INC  IX
43F0 FE 22      1566            CP   '"'
43F2 28 D1      1567            JR   Z,T_SKIP
43F4 18 F5      1568            JR   _STRSKIP
43F6            1569 ;
43F6            1570 __GOSUB
43F6 ED 73 E4   1571            LD   (STK_WR),SP
43F9 46
43FA ED 7B E2   1572            LD   SP,(RET_SP)
43FD 46
43FE DD E5      1573            PUSH IX
4400 ED 73 E2   1574            LD   (RET_SP),SP
4403 46
4404 ED 7B E4   1575            LD   SP,(STK_WR)
4407 46
4408 C3 80 43   1576            JP   __GOTO
440B            1577 @_IF
440B E1         1578            POP  HL
440C 7D         1579            LD   A,L
440D B4         1580            OR   H
440E 28 03      1581            JR   Z,_IF_SKIP
4410 C3 E3 42   1582            JP   CCMAIN
4413            1583 _IF_SKIP
4413 CD C5 43   1584            CALL T_SKIP
4416 C3 E3 42   1585            JP   CCMAIN

                1586 ;                  DEBUG SUPORT J.YAMADA
                1587 COM_TBL
                1588 ;  00-0FH
                1589        DB    0, 0, 0, 0   ; . . . .
                1590        DB    0, 0, 0,129  ; . . . VAL$
                1591        DB    0, 0, 0,89   ; . . . INP$
                1592        DB  124, 0, 0, 6   ; L/MD . . /MOD
                1593 ;  10-1FH
                1594        DB    0, 0,123, 0  ; . . CMP2 .
                1595        DB   87,88, 0,22   ; VAL1 VAL2 . HEX2
                1596        DB    0,23,83, 0   ; . HEX4 PRF2 .
                1597        DB   39, 0, 0,31   ; CALL . . BELL
                1598 ;  20-2FH
                1599        DB   54,41, 0, 0   ; HIGH GETA . .
                1600        DB   43, 0, 0,92   ; GETD . . FILL
                1601        DB   45,103, 3, 1  ; GETH BOX@ * +
                1602        DB   74, 2,104, 4  ; LMOD - TILE /
                1603 ;  30-3FH
                1604        DB    0,86,100, 0  ; . HEXL WIND .
                1605        DB   96,14,21, 0   ; INIT DROP SCRN .
                1606        DB   27,33,40,16   ; COTR GOTO PUTA COPY
                1607        DB    8,42, 9,111  ; < PUTD > PRON
                1608 ;  40-4FH
                1609        DB    0,44,60,61   ; . PUTH CURX CURY
                1610        DB    0,85, 0, 0   ; . STRL . .
                1611        DB  113,26, 0,126  ; PEEK# PRTS . *!
                1612        DB    0, 0, 0,64   ; . . . LEFT$
                1613 ;  50-5FH
                1614        DB   84, 0,114,125; STRW . POKE# D/MOD
                1615        DB    0, 0,105, 0  ; . . TRIANGLE .
                1616        DB    0, 0, 0,53   ; . . . LOOP!
                1617        DB    0, 0,10, 0   ; . . != .
                1618 ;  60-6FH
                1619        DB    0,110, 0, 0  ; . MAGIC . .
                1620        DB    0,130, 0,46  ; . BREAK . PEEKB
                1621        DB  101,77, 0, 0   ; LINE@ ASCII . .
                1622        DB   15,79,75,30   ; SWAP1 =0 D* WIDCH
                1623 ;  70-7FH
                1624        DB    0,48,19, 0   ; . POKEB FLGET .
                1625        DB    0, 0,72,70   ; . . L* L+
                1626        DB    0,71, 7,73   ; . L- == L/
                1627        DB   47,112, 0,95  ; PEEKW PROFF . SWAPD
                1628 ;  80-8FH
                1629        DB   34,94,78, 0   ; GOSUB DROPL F< .
                1630        DB    0, 0,49,93   ; . . POKEW COPYL
                1631        DB  116,117,108,0  ; I? J? POINT .
                1632        DB   38,24, 0,36   ; UNTIL PRINT . IF
                1633 ;  90-9FH
                1634        DB   68, 0, 0,52   ; INSTR . . DO
                1635        DB    0,118, 0,50  ; . CR . IN
                1636        DB  121, 0, 0, 0   ; FR . . .
                1637        DB    0,56, 0, 0   ; . EX . .
                1638 ;  A0-AFH
                1639        DB    0,12,65, 0   ; . OR RIGHT$ .
                1640        DB    0, 0,120,0   ; . . TR .
                1641        DB    0, 0, 0, 0   ; . . . .
                1642        DB    0, 0, 0, 0   ; . . . .
                1643 ;  B0-BFH
                1644        DB    0, 0,107,90  ; . . CIRCLE TRANS+
                1645        DB   62,91,99, 0   ; NEGATE TRANS- PALET@ .
                1646        DB   32, 0, 0,102  ; LOCATE . . SLINE@
                1647        DB    0, 0,80,81   ; . . PRINT1 PRINT2
                1648 ;  C0-CFH
                1649        DB    0,37, 0, 0   ; . REPEAT . .
                1650        DB    0, 0, 0, 0   ; . . . .
                1651        DB    0,18, 0, 0   ;  . GETKEY . .
                1652        DB   29, 0, 0, 0   ; DEC . . .
                1653 ;  D0-DFH
                1654        DB  106,66,122,11  ; BOXFUL STRCAT LEA AND
                1655        DB    0, 0, 0,115  ; . . . END
                1656        DB   67,69,28, 0   ; STRLEN STRCMP INC .
                1657        DB    0,25,97, 0   ; . CHR COL .
                1658 ;  E0-EFH
                1659        DB    5, 0,98,76   ; MOD . CLS CTL
                1660        DB   20,63, 0,109  ; RND STRCPY . DOT
                1661        DB   82,17, 0,35   ; PRF KEY . RET
                1662        DB    0,59, 0,128  ; . ROL . DEC#
                1663 ;  F0-FFH
                1664        DB    0,57,55,58   ; . NOT LOW ROR
                1665        DB    0,119, 0, 0  ; . ROT . .
                1666        DB   51,13, 0, 0   ; OUT XOR . .
                1667        DB    0,127,131,0  ; . INC# MID$ @
                1668 JUMPTBL
                1669        DW   ERROR1
                1670        DW   @TASU
                1671        DW   @HIKU
                1672        DW   @MLT
                1673        DW   @DIV
                1674        DW   @MOD
                1675        DW   @DIVMOD
                1676        DW   @==
                1677        DW   @<
                1678        DW   @>
                1679        DW   @!=
                1680        DW   @AND
                1681        DW   @OR
                1682        DW   @XOR
                1683 ;
                1684        DW   @DROP
                1685        DW   @SWAP1
                1686        DW   @COPY
                1687 ;
                1688        DW   @KEY
                1689        DW   @GETKEY
                1690        DW   @FLGET
                1691        DW   @RND
                1692        DW   @SCRN
                1693 ;
                1694        DW   @HEX2
                1695        DW   @HEX4
                1696        DW   @PRINT
                1697        DW   @CHR
                1698        DW   @PRTS
                1699        DW   @COTR
                1700 ;
                1701        DW   @INC
                1702        DW   @DEC
                1703        DW   @WIDCH
                1704        DW   @BELL
                1705        DW   @LOCATE
                1706 ;
                1707        DW   @GOTO
                1708        DW   @GOSUB
                1709        DW   @RET
                1710        DW   @IF
                1711        DW   @REPEAT
                1712        DW   @UNTIL
                1713 ;
                1714        DW   @CALL
                1715        DW   @PUTA
                1716        DW   @GETA
                1717        DW   @PUTD
                1718        DW   @GETD
                1719        DW   @PUTH
                1720        DW   @GETH
                1721 ;
                1722        DW   @PEEKB
                1723        DW   @PEEKW
                1724        DW   @POKEB
                1725        DW   @POKEW
                1726 ;
                1727        DW   @IN
                1728        DW   @OUT
                1729 ;
                1730        DW   @DO
                1731        DW   @LOOP!
                1732 ;
                1733        DW   @HIGH
                1734        DW   @LOW
                1735        DW   @EX
                1736        DW   @NOT
                1737        DW   @ROR
                1738        DW   @ROL
                1739        DW   @CURX
                1740        DW   @CURY
                1741        DW   @NEGATE
                1742 ;
                1743        DW   @STRCPY
                1744        DW   @LEFT$
                1745        DW   @RIGHT$
                1746        DW   @STRCAT
                1747        DW   @STRLEN
                1748        DW   @INSTR
                1749        DW   @STRCMP
                1750 ;
                1751        DW   @LTASU
                1752        DW   @LHIKU
                1753        DW   @LMLT
                1754        DW   @LDIV
                1755        DW   @LMOD
                1756        DW   @DMLT
                1757        DW   @CTL
                1758 ;
                1759        DW   @ASCII
                1760        DW   @F<
                1761        DW   @=0
                1762 ;
                1763        DW   @PRINT1
                1764        DW   @PRINT2
                1765        DW   @PRF
                1766        DW   @PRF2
                1767 ;
                1768        DW   @STRW
                1769        DW   @STRL
                1770        DW   @HEXL
                1771        DW   @VAL1
                1772        DW   @VAL2
                1773        DW   @INP$
                1774        DW   @TRANS1
                1775        DW   @TRANS2
                1776        DW   @FILL
                1777        DW   @COPYL
                1778        DW   @DROPL
                1779        DW   @SWAPD
                1780 ;
                1781        DW   @INIT
                1782        DW   @COL
                1783        DW   @CLS
                1784        DW   @PALET
                1785        DW   @WIND
                1786        DW   @LINE
                1787        DW   @SLINE
                1788        DW   @BOX
                1789        DW   @TILE
                1790        DW   @TRIANGLE
                1791        DW   @BOXFUL
                1792        DW   @CIRCLE
                1793        DW   @POINT
                1794        DW   @DOT
```

▶ドラクエⅣを買ったのだが，さすがである。パソコンソフトも見習わなければいけないだろう。しかし，最後のボスはかわいそうであった。救われないならないで極悪なボスにすればよかったと思ってしまった。　　吉村　昇(18) X68000 大阪府

```
1795            DW    @MAGIC
1796 ;
1797            DW    @PRON
1798            DW    @PROFF
1799            DW    @PEEK#
1800            DW    @POKE#
1801            DW    END
1802            DW    @I?
1803            DW    @J?
1804            DW    @CR
1805            DW    @ROT
1806 ;
1807            DW    @TR
1808            DW    @FR
1809            DW    @LEA
1810            DW    @CMP2
1811 ;
1812            DW    @LDIVMD
1813            DW    @DDIVMOD
1814            DW    @MLT!
1815            DW    @INC#
1816            DW    @DEC#
1817            DW    @VAL$
1818            DW    @BREAK
1819            DW    @MID$
1820 ; RESERVE
1821            DW    0
1822            DW    0
1823            DW    0
1824            DW    0
1825 ;
1826            DW    @_LABEL
1827            DW    @_IF
```

```
1828            DW    STRING
1829            DW    @_GOTO
1830            DW    @_GOSUB
1831            DW    @_LETW
1832            DW    @_HENW
1833            DW    @_WORDT
1834            DW    @_INC
1835            DW    @_DEC
1836            DW    @_LETL
1837            DW    @_HENL
1838            DW    @_LONGT
1839 ;
1840 TOPDATA
1841            DB    0
1842            DM    "+-*/M/=<>!AOXDSC"
1843            DM    "KGFRSHHPCPCIDWBL"
1844            DM    "GGRIRUCPGPGPGPPP"
1845            DM    "PIODLHLENRRCCNSL"
1846            DM    "RSSISLLLLLDCAF=P"
1847            DM    "PPPSSHVVITTFCDSI"
1848            DM    "CCPWLSBTTBCPDMPP"
1849            DM    "PPEIJCRTFLCLD*ID"
1850            DM    "VBM"
1851            DB    0,0,0,0
1852 ;
1853 ; STACK  WORK AREA
1854 ;
1855 CPUSTK
1856            DW    0
1857 LINE_WR
1858            DW    0
1859 LINE_WR1
1860            DW    0
```

```
1861 LINE_TOP
1862            DW    0
1863 RND0
1864            DW    0
1865 CFLAG
1866            DB    0
1867 VAR1
1868            DW    0
1869 VAR2
1870            DW    0
1871 #HL
1872            DW    0
1873 #DE
1874            DW    0
1875 #A
1876            DB    0
1877 ;
1878 CCFLAG
1879            DB    0
1880 CCTEXT
1881            DW    $8000
1882 RET_SP
1883            DW    @RET_SP
1884 STK_WR
1885            DW    0
1886 DIRECT_B
1887            DS    256
1888 VAR
1889            DS    520
1890 LABELT
1891            DS    1024
```

全機種共通システムインデックス

■85年 6 月号
序論　共通化の試み
第 1 部　S-OS"MACE"
第 2 部　Lisp-85インタプリタ
第 3 部　チェックサムプログラム
■85年 7 月号
第 4 部　マシン語プログラム開発入門
第 5 部　エディタアセンブラZEDA
第 6 部　デバッグツールZAID
■85年 8 月号
第 7 部　ゲーム開発パッケージBEMS
第 8 部　ソースジェネレータZING
■85年 9 月号
インタラプト　S-OS番外地
第 9 部　マシン語入力ツールMACINTO-S
第10部　Lisp-85入門(1)
■85年10月号
第11部　仮想マシンCAP-X85
連載　Lisp-85入門(2)
■85年11月号
連載　Lisp-85入門(3)
■85年12月号
第12部　Prolog-85発表
■86年 1 月号
第13部　リロケータブルのお話
第14部　FM音源サウンドエディタ
■86年 2 月号
第15部　S-OS"SWORD"
第16部　Prolog-85入門(1)
■86年 3 月号
第17部　magiFORTH発表
連載　Prolog-85入門(2)
■86年 4 月号
第18部　思考ゲームJEWEL
第19部　LIFE GAME
連載　基礎からのmagiFORTH
連載　Prolog-85入門(3)
■86年 5 月号
第20部　スクリーンエディタE-MATE
連載　実戦演習magiFORTH
■86年 6 月号
第21部　Z80TRACER
第22部　magiFORTH TRACER
第23部　ディスクダンプ&エディタ
第24部　"SWORD" 2000 QD
連載　対話で学ぶ magiFORTH
特別付録　PC-8801版S-OS"SWORD"
■86年 7 月号
第25部　FM音源ミュージックシステム
付録　FM音源ボードの製作
連載　計算力アップのmagiFORTH
特別付録　SMC-777版 S-OS"SWORD"
■86年 8 月号
第26部　対局五目並べ
第27部　MZ-2500版 S-OS"SWORD"
■86年 9 月号
第28部　FuzzyBASIC 発表
連載　明日に向かって magiFORTH
■86年10月号
第29部　ちょっと便利な拡張プログラム
第30部　ディスクモニタ DREAM
第31部　FuzzyBASIC 料理法<1>

■86年11月号
第32部　パズルゲーム HOTTAN
第33部　MAZE in MAZE
連載　FuzzyBASIC 料理法<2>
■86年12月号
第34部　CASL & COMET
連載　FuzzyBASIC 料理法<3>
■87年 1 月号
第35部　マシン語入力ツールMACINTO-C
連載　FuzzyBASIC 料理法<4>
■87年 2 月号
第36部　アドベンチャーゲーム MARMALADE
第37部　テキアベ作成ツール CONTEX
■87年 3 月号
第38部　魔法使いはアニメがお好き
第39部　アニメーションツール MAGE
付録　"SWORD" 再掲載と MAGIC の標準化
■87年 4 月号
第40部　INVADER GAME
第41部　TANGERINE
■87年 5 月号
第42部　S-OS"SWORD" 変身セット
第43部　MZ-700用 "SWORD" を QD 対応に
■87年 6 月号
インタラプト　コンパイラ物語
第44部　FuzzyBASIC コンパイラ
第45部　エディタアセンブラ ZEDA-3
■87年 7 月号
第46部　STORY MASTER
■87年 8 月号
第47部　パズルゲーム碁石拾い
第48部　漢字出力パッケージ JACKWRITE
特別付録　FM-7/77版 S-OS"SWORD"
■87年 9 月号
第49部　リロケータブル逆アセンブラ Inside-R
特別付録　PC-8001/8801 版 S-OS"SWORD"
■87年10月号
第50部　tiny CORE WARS
第51部　FuzzyBASIC コンパイラの拡張
第52部　X1turbo 版 S-OS"SWORD"
■87年11月号
序論　神話のなかのマイクロコンピュータ
付録　S-OS の仲間たち
第53部　もうひとつの FuzzyBASIC 入門
第54部　ファイルアロケータ&ローダ
インタラプト　S-OS こちら集中治療室
第55部　BACK GAMMON
■87年12月号
第56部　タートルグラフィックパッケージTURTLE
第57部　X1turbo 版 "SWORD" アフターケア ラインプリントルーチン
特別付録　PASOPIA7 版 S-OS"SWORD"
■88年 1 月号
第58部　FuzzyBASIC コンパイラ・奥村版
付録　石上版コンパイラ拡張部の修正
■88年 2 月号
第59部　シューティングゲーム ELFES
■88年 3 月号
第60部　構造型コンパイラ言語 SLANG
■88年 4 月号
第61部　デバッギングツール TRADE
第62部　シミュレーションウォーゲーム WALRUS

■88年 5 月号
第63部　シューティングゲーム ELFES II
第64部　地底最大の作戦
■88年 6 月号
第65部　構造化言語 SLANG 入門(1)
第66部　Lisp-85 用 NAMPA シミュレーション
■88年 7 月号
第67部　マルチウィンドウドライバ MW-1
連載　構造化言語 SLANG 入門(2)
■88年 8 月号
第68部　マルチウィンドウエディタ WINER
■88年 9 月号
第69部　超小型エディタ TED-750
第70部　アフターケア WINER の拡張
■88年10月号
第71部　SLANG 用ファイル入出力ライブラリ
第72部　シューティングゲーム MANKAI
■88年11月号
第73部　シューティングゲーム ELFES IV
■88年12月号
第74部　ソースジェネレータ SOURCERY
■89年 1 月号
第75部　パズルゲーム LAST ONE
第76部　ブロックゲーム FLICK
■89年 2 月号
第77部　高速エディタアセンブラ REDA
特別付録　X1版 S-OS"SWORD"<再掲載>
■89年 3 月号
第78部　Z80用浮動小数点演算パッケージSOROBAN
■89年 4 月号
第79部　SLANG 用実数演算ライブラリ
■89年 5 月号
第80部　ソースジェネレータ RING
■89年 6 月号
第81部　超小型コンパイラTTC
■89年 7 月号
第82部　TTC用パズルゲーム　TICBAN
■89年 8 月号
第83部　CP/M用ファイルコンバータ
■89年 9 月号
第84部　生物進化シミュレーションBUGS
■89年10月号
第85部　小型インタプリタ言語TTI
■89年11月号
第86部　TTI用パズルゲーム PUSH BON!
■89年12月号
第87部　SLANG用リダイレクションライブラリ DIO. LIB
■90年 1 月号
第88部　SLANG用ゲームWORM KUN
特別付録　再掲載SLANGコンパイラ
■90年 2 月号
第89部　超小型コンパイラTTC++
■90年 3 月号
第90部　超多機能アセンブラOHM-Z80
■90年 4 月号
第91部　ファジィコンピュータシミュレーションI-MY

＊以上のアプリケーションは，基本システムであるS-OS "MACE" またはS-OS"SWORD" がないと動作しませんのでご注意ください。

▶何の勉強もせずに，大学に受かったが，三流（五流という話もある）大学なので，なにも言えない。
田中　順次（18）X1turbo II　大阪府

X68000の新しいビジュアル環境

これがSX-WINDOWだ!

Yoshida Kouichi
吉田 幸一

X68000がついにウィンドウシステムを搭載した。SX-WINDOWの登場である。Oh!Xでは2回にわたってこのウィンドウ環境がもたらす世界を追ってみたい。まず今回はコマンドシェルに代わるオペレーションの基本を見ていくことにしよう。

真実のところ，シャープからX68000用ウィンドウシステムが出るという話を聞いてとても不安であった。なによりも，**美的センスのカケラもないケバいMS-WINDOWS**やOS/2のPM（プレゼンテーションマネージャといって，つまりはウィンドウシステムのこと）に似てたらどうしよう，Macintoshに似てて訴えられたらどうしよう，OS-9やX-Windowみたいにテキストのシェルがたくさん開くようなのだったらどうしよう，という**似てたらどうしよう**症候群だったのである。さらに，“オリジナルであることだけを売り物にした悲惨な出来だったらどうしよう”不安もあった。しかし，だ。ひと目見た瞬間，これらの不安は光年の彼方へとんでいった。

カッコいいじゃん！

とまあ，そういうことだ。どれにも似てなくてカッコいい。これを見てX68000が欲しくなったという98ユーザーがいたくらいカッコいい。こんないいことはない。

強いていえば，色使いやアイコンの雰囲気が“NeXT”に似ている。あのスティーブ・ジョブズの68030マシン，NeXTである。基本的には全然違うが，とりあえず，雰囲気は似ている。

SX-WINDOWの使い方

SX-WINDOWはとてもセンスのよいユーザーインタフェイスを持っている。そいつを実際にご覧にいれよう。

まず，写真のように立ち上がる。ウィンドウシステムだけあって，いろんなウィンドウが開くのである。

見てわかるとおりメニューバーはない。**プルダウンメニュー形式ではない**ということだ。ビジュアルシェル同様，ポップアップメニューなのである。脱プルダウン（そういえばNeXTもそうだった）。異論もあるだろうが，私はプルダウンメニューが万能とは思わない。こういったシステムでは操作の対象となるオブジェクトの位置で機能を選択できるポップアップメニューのほうが有効だろう（画面のいちばん上までいちいちマウスカーソルを運ぶのは面倒）。

ファイルウィンドウを開く

まずはファイルのウィンドウである。例によってタイトルバーがあり，スクロールバーやらウィンドウのサイズを変えるボタンやらがある。ファイルウィンドウの触り方だが，タイトルバーにはドライブとディレクトリ名が書いてある。こいつを左クリックでドラッグするとウィンドウが移動するのは慣れたもの。大筋はビジュアルシェルと同じである。

注目すべきはタイトル部左右の**クリップ**と◀だ。クリップをクリックすると，ウィンドウがクリップで止められた姿になる。こうすると，子ディレクトリを開いたとき，親ディレクトリは自動的にクローズされるのだ。ディレクトリをクリップで束ねたというイメージだろうか。何がいいかというと，ハードディスクなどで深い階層ディレクトリを追いかけるとき，画面がウィンドウの洪水にならないのである。ウィンドウシステムの場合，やたらとウィンドウが開くので，こういった機能はおいしい。さらに◀は“CD..”の機能をウィンドウに持たせたわけで，非常に便利である。世間のウィンドウには“親ディレクトリアイコン”を持たせることで解決しているものもあったが，このほうがずっとスマートだ。

・起動時にはマウスカーソルの右と左が交互に赤く点滅。なんと，ちょっと待ってタイムには，踏切のあの点滅する信号になるのだ。ビジュアルシェルでは砂時計で，サイバーノートでは柱時計だったんだけど，どちらでもない，新しくて誰も気づかなくて誰が見てもそのとおりな踏切を採用したアイデアには脱帽だ。
・ドライブ名の左にあるのがクリップ，右の◀が親ディレクトリへ戻るアイコン，右上のバツがクローズアイコンである。
・ちょっと見慣れないのがスクロールバー。上向き矢印をクリックすると，ウィンドウ内の表示は下から上に，スクロールボックスは逆に上から下にという複雑な動作をする。
・右下のウィンドウサイズボタンは，普通にドラッグすればウィンドウの拡大縮小ができ，ダブルクリックすると一気にウィンドウが画面一杯に拡がるという寸法なのであった。

▶某店の広告に，「○○（店名）は内税で頑張りますので，消費税は上積みしません」とある。なーにが「だから」だ。内税に上積みしたらサギだ。
松本 太（19）X1G/turboZ 大阪府

さて，ビジュアルシェル同様，右クリックでポップアップメニューが開く。**ウィンドウ情報**とか**すべてを選択**とか**ワイルドカード，整頓，新規ディレクトリ，クローズ**とかがある。字が小さくワークステーションみたいでオシャレなのは相変わらずだ。

ウィンドウ情報はルートディレクトリならドライブ情報が，そうでないなら，そのディレクトリの情報が出る。ここでのチェックポイントは，“ウィンドウ情報のウィンドウが開く”ということにつきるだろう。つまり，ウィンドウ情報を動かしたり，ウィンドウ情報を開いたまま次の作業ができたりするのである。その上，**文字が壁に彫ったみたいで，カッコいい**。

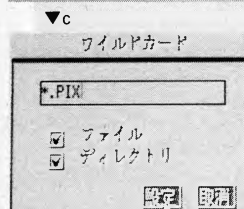
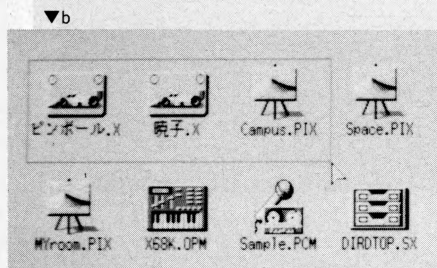
続いて**すべてを選択**というのは，ウィンドウ内のすべてのファイルやディレクトリを選択することである（選択というキーワードが出てきたが，その話は次でする）。

で，上に挙げたなかにはソートがなかった。ソートは，SHIFTキーを押しながら右クリックなのだ。すると，5種類くらいのソートメニューが出るので，ドラッグして選ぼう。拡張子順なんてのは便利そうだ。

例によってファイルアイコンやディレクトリアイコンをダブルクリックすると実行したり，このファイルは実行できません，ピンポンとなったり，ディレクトリならばそれが開いたりする。1回だけのクリックならば，アイコンが反転する。その状態でマ

▼a

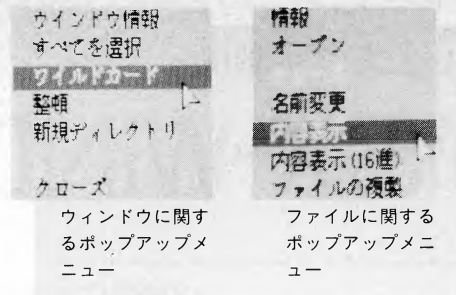
▼b

▼c

a. SHIFT＋左クリックでファイルの複数選択
b. ボックスラインで囲んで選択
c. ワイルドカードも使える

ウインドウ情報
すべてを選択
ワイルドカード
整頓
新規ディレクトリ
クローズ

ウィンドウに関するポップアップメニュー

情報
オープン
名前変更
内容表示
内容表示(16進)
ファイルの複製

ファイルに関するポップアップメニュー

ウスカーソルをアイコンの上に置いて右クリックすると，またもやお馴染みの**情報，オープン，クローズ，名前変更，内容表示，内容表示(16進)，ファイルの複製**だ。ファイル情報もウィンドウ情報と同様に情報ウィンドウが開く。ここでは名前や容量やファイルの属性（ATTRIBコマンドで設定する，読み出し専用とか不可視だとか）を設定できたりもする。ファイル情報ウィンドウがほかのウィンドウと同様にいくつも開けるため，2つのファイルの情報を開いて，比較するなんてのも可能だ。

ここでの要チェックは**内容表示**と**ファイル複製**だろう。**内容表示**は待望のTYPEコマンドなのであった。これを選ぶと**TYPEウィンドウ**が開いて，中にずらずらとタイプされるのだ。もちろん，スクロールしたり印刷したりできるのだ。ただスクロールやテキスト表示速度が遅いのが難だけどね。ついでに，**内容表示(16進)**というのはもうわかるとおり，**DUMPウィンドウ**が開くのであった。ウィンドウばしばし。

ファイル複製というのは，ビジュアルシェルのコピー機アイコンがなくなった代わりに設けられたものだ。なぜなら，ファイルアイコンを別のウィンドウに移動させたとき，**別ドライブならCOPY，同じドライブの別ディレクトリならMOVEする**のが基本となったからだ。うーむ。完璧。

ついでに，名前変更やファイル複製のファイル名指定時には，大文字化・小文字化ボタンがあって，それをクリックするとCASEするのであった。あっぱれ。

内容表示。TYPE用のウィンドウが開く

さらに細かくファイル操作を見る

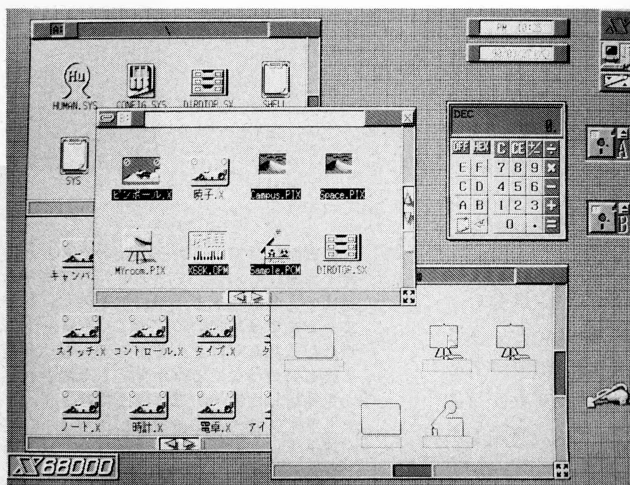
とりあえず，ウィンドウシステムに欠かせないファイル操作である。ビジュアルシェルと比較していただきたい。

2つのウィンドウ間でファイルアイコンをドラッグすれば前に書いたようにコピーしたりムーブしたりする。が，ビジュアルシェルではひとつずつしかできなかった。SX-WINDOWではそんなことはない。複数のファイルをコピーしたいとき，3つも方法があるのだ。

ひとつは，**SHIFTキー併用による追加選択**だ。この方法はKamikazeでも，Macintoshでも採用している世界標準である。

2番目の方法として，ウィンドウ上の適当な位置で左クリックし，そのままドラッグするのだ。そうすると，グラフィックソフトのようなボックスラインが出てくるので，そいつで**選択したいファイルアイコンを囲んでしまえ**である。囲んだ部分に不要なファイルがあれば，SHIFT＋左クリックでそいつだけ解除，囲った以外にも欲しい

複数のファイルをいっぺんにMOVE（コピー）

▶ベルリンの壁，こんなもん買うやつアホや！　でも，貰えるんやったら，欲しいなぁ。
松下　耕三（21）X1turboZ　大阪府

ものがあれば，SHIFT＋左クリックで追加してやればいい。なんてこった。

3つ目は，ウィンドウ上でのポップアップメニューから**ワイルドカードを選んで，＊.DOCとかすればいい**のだ。すると，.DOCのついたものが全部選択される。しかも，ワイルドカード指定ではファイルかディレクトリかの選択もできる。これはDIRコマンドよりいいぞ。

さて，ファイルを選択した。次はファイルのコピーやらムーブやら削除である。たくさんあるファイルのどれかをドラッグしてみよう。すると，全部のファイルアイコンの枠がずりずりと一緒についてくるではないか。これは面白い，ということで，画面引き回しの刑に処したあと，目的のウィンドウ上で止める。すると，ファイルのコピー（あるいはムーブ）が始まる。親切にもメッセージが開いて，いま何個目をコピー（あるいはムーブ）してるかを表示，途中で中止できたりもする。

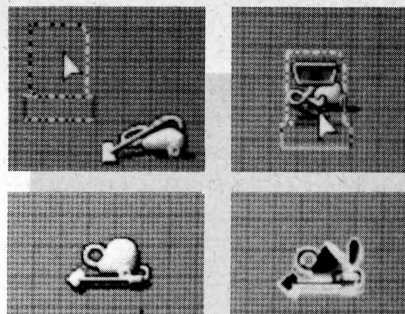
削除の場合は，**クリーナー**アイコンだ。これも重要ね。Macintoshやビジュアルシェルはごみ箱だった。NeXTはブラックホールであった。なんと，X68000は，並のセンスではなかった，家電メーカーらしく**電気掃除機**なのだ！

ちなみに，ビジュアルシェルではごみ箱はTRASHというディレクトリに過ぎなかったが，SX-WINDOWでは，クリーナーに放り込んだ時点ではデリートするファイルやディレクトリの情報を保持してファイルアイコンを消すだけで，実際の削除はクリーナーを空にしたときに行われる。だから，クリーナーのポップアップメニューには元に戻すがあって，それが簡単に実現できるのだ。

ドライブ管理で遊ぶ

以上なわけでファイル操作はできるのであるが，ファイルを操作するには，そのファイルの入っているドライブが必要である(当たり前だ)。

まず，ドライブアイコンは画面右にある。5インチの場合はアイコンの右上隅にイジェクトボタンがあったりする。光磁気ディスクアイコンの用意も万端だ。

ドライブアイコン上で右ボタンを押すと，ご想像どおり，ポップアップメニューが開く。**ドライブ情報，オープン，クローズ，名前変更，フォーマット**とくる。フォーマットの前には1行分の空白があるのがまたユーザーフレンドリーである。

ファイルをクリーナーに重ねると吸い込み口はこっちを向くわ，クリーナーにファイルを放り込むと膨らむわの大サービス，ニューライフピープルである

ドライブのコピーは，お馴染みの**ドライブアイコンをドラッグして重ねろ，**である。すると，小さなウィンドウが現れる。なんと，**ディスクコピー**と**コピーオール**が選択できるのだ。コピーオールは，コピー先のファイルはそのままに，ファイルが全部コピーされる便利なもの（オプションなしのCOPYALLと同じ)。なんともはや。

ところで，このドライブというやつは意

ウィンドウシステムとは

SX-WINDOWの話を始める前に，ウィンドウシステムの話を忘れてはいけない。なんでウィンドウシステムなのか，ビジュアルシェルとどう違うのかってことだ。

ウィンドウシステムというのは，ただマルチウィンドウだったりすればいいわけじゃあない。システムというだけあって，X68000に関する全操作がその上でできるようでなければならない。たとえば，Macintoshにコマンドシェルはないが誰も文句はいわない（いうのは Macintosh をセカンドマシンに買ったMS-DOSユーザーだろう)。

従来のビジュアルシェル（VS.X）がウィンドウシステムでなく単なる“ファイルハンドラ”だというのは，ビジュアルシェル上ではファイル操作とプログラムの実行以外はできなかったからである。つまり Human68k 上のアプリケーションのひとつにすぎない。ビジュアルシェルからプログラムを実行してもビジュアルシェル上で動いているとはいわないのはそういうわけだ。

ウィンドウシステムというからには，コマンドシェルが，Human68kにコマンドを使ってマシンをコントロールするという環境を提供していたのと同様に，マルチウィンドウを使って X68000 をコントロールする環境を備えてなければならない。command.x が Human68k にコマンドやファイル入出力の機能を付加したように，ウィンドウシステムは Human68k にウィンドウの機能を付加しなければならないということだったりする。

これは結構大変なことである。Macintoshが1984年に発売されて以来，ウィンドウシステムの評価は高まってきているにもかかわらず，**Macintosh を超えるパソコン用ウィンドウシステムが出てこなかった**のを見てもわかる。

ウィンドウシステムの条件

じゃあ，ウィンドウシステムに最低限必要なものは何か。そこいらへんを押さえてないと，結局みんなcommand.xに戻っていきましたとさ，おしまい，になってしまう。

ひとつ。ウィンドウシステムはそのシステム上でパソコンのコントロールに必要なすべての操作ができなければならない。あるいはそういったコントロールが可能なコマンドを作成できなければならない。

ひとつ。ウィンドウシステムは原則として専用に書かれた全アプリケーションをメモリ不足という理由以外ではその環境で走らせることができるキャパシティを持っていなければならない。

ひとつ。ウィンドウシステムはそのシステム上で動くプログラムの開発を支援する機構（ウィンドウ機能を実現するための書式やライブラリ，ツールなど）を持っていなければならない。

ひとつ。ウィンドウシステムは開いているどのウィンドウも差別してはならない。

ひとつ。ウィンドウシステムはアプリケーション間でデータをやりとりする術を持っていなければならない。

ひとつ。ウィンドウシステムはセンスがよくなければならない。

まだあるような気がするけど，とりあえずこんなもんで，**一番重要なのはセンス**だな。なぜかっていうと，ウィンドウシステムが目指しているのはマシンをよりユーザーフレンドリーなものにすることだからだ。そして次は，ウィンドウ上のシステムを開発するための**環境**だ。これがないと，誰もウィンドウ上のプログラムなど書いてはくれない。ウィンドウシステム上では(考えればわかることだけど)，そのウィンドウ用に開発されたプログラムでないと実行できないのだ。

で，SX-WINDOWはというと，正しくウィンドウシステムなのであった。

従来のXファイルはどーするんだ

ウィンドウシステムはウィンドウシステム用に書かれたアプリケーションしか走らない。では，蓄積された膨大なプログラムはどーするんだ，てな問題は出てくる。

SX-WINDOWでは，ウィンドウ対応でない実行ファイルに対しては一度ウィンドウを終了してから COMMAND.X 上で実行するというふうになっているので，心配はない。これは**ビジュアルシェルからプログラムを起動したときと同じ感覚**である。現にいまだって，そうやって原稿を書いているのだ。まだまだウィンドウ対応のコマンドやプログラムは（ほとんど）ないので，それでいいのだ。ただし，ウィンドウのためのデバイスドライバが常駐してたりウィンドウのカーネルは残っているので，メモリ的なハンディは当然ある。

ちなみに，ウィンドウからチャイルドプロセスを実行すると，現時点でのかなりのウィンドウ情報を失うので（その代わり，フリーエリアは広くなる)，スタート画面設定で終了時画面を登録にしておかない限り，EXITで戻っても元の姿には戻ってくれないので注意。

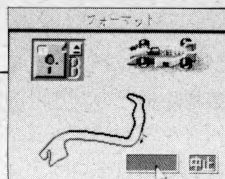
▶ Just a Ronin！　　国政　寛（18）X68000，PC-6001　大阪府

外と厄介なもので，私のようにハードディスクを５つもの領域に分けていたりすると，５インチ×２＋ハードディスク×５でなわけで，Ｇドライブまで必要である。ビジュアルシェルでは５番目以降は矢印で選択していたが，それはスマートではない。でも，つながっているドライブの全部を表示するのもうっとうしい。

ところがどっこい，SX-WINDOWを作った人は賢いのであった。

まず，ドライブアイコンであるが，そのほかのウィンドウと同じく，好きな場所に置けるので，画面の真ん中や左上なんかに左遷してもよい。それでもドライブアイコンが７つも８つもあれば，いくら小さなアイコンとはいえ場所をとる。ここからが佳境である。そんなとき，右上のＸと書いてあるシステムアイコンで右ボタンを押し，ポップアップメニューから**ドライブトレイ**を選択する。すると，ドライブトレイウィンドウがでろんと開く。こいつがポイントだ。ドライブトレイというのは，X68000につながっている全ドライブアイコンがのっかっている，ドライブ管理ウィンドウなのである。

たとえば，辞書専用のドライブは普段は使わないので，ドライブトレイにしまっておけば邪魔にならないのである。ドライブトレイからでもダブルクリックすればそいつを開くことはできるので，問題はない。ただし，ドライブトレイにあるフロッピーディスクはイジェクトできないとか，右ボタンのポップアップメニューが開かないといったハンディはある。ドライブトレイにしまうには画面のドライブアイコンをドライブトレイにドラッグすればよく，ドライブトレイから出すには，その逆だ。簡単。

右上の３アイコンとポップアップメニュー

なんとフォーマットの際にはF1マシンと，どっかのサーキットの絵が出るのである。スタートすると，フォーマットが進むにつれて，コース上が赤く塗られていく。１周するとフォーマット終了で，F1マシンの絵の上にチェッカーフラッグまで出るのであった。おお，はらほろひれはれ。

ドライブトレイにドライブを収納

つまり，SX-WINDOWはウィンドウをばしばし開く代わりに，余計なアイコンやウィンドウを隠しておく機能にも優れているのだ。なんでもできるように見えて，無法地帯ではないという，理想郷のように平和な世界なのだ。

SX-WINDOWの機能―その他編

さて，ファイルやドライブの扱い方しか見てはこなかったが，そのほかにも，右上の３アイコンにはポップアップメニューが隠されている。そいつを紹介しておこう。

まずＸのシステムアイコンであるが，これには**ドライブトレイ**のほかに，**シェル情報**，**プロセス情報**，**スタート画面設定**，**全クローズ**，**終了**がある。

シェル情報というのは，Human68kとSX-WINDOWとSX-WINDOW用ツールのバージョンを表示するものである。

プロセス情報というのは，空きメモリの状態（MEMFREEコマンドに似ている）を表示するものである。

スタート画面がこれまた，フレンドリーな機能である。立ち上げ時に開くウィンドウなんかの設定ができるのである。ビジュアルシェルみたいに，いきなりつながっているドライブを全部開いたりはしない。現在の画面を登録するか，終了時の画面を登録するかを選択できて，便利である。

全クローズはいいとして，**終了**を選択すると，SX-WINDOWのタイトルが出て，それが赤くなって，止まる。そうなったらリセットか電源OFFだな。

Ｘの下のX68000アイコンはアクセサリーアイコンである。ポップアップメニューで**ノート**やら**電卓**やら**カレンダー**やら**時計**やらといろいろあるが，それらの説明は来月の**コマンド/アプリケーション編**で紹介しよう。なぜなら，クリップボードを除いて，どれもメニューに応じたファイルを実行するだけだからである。

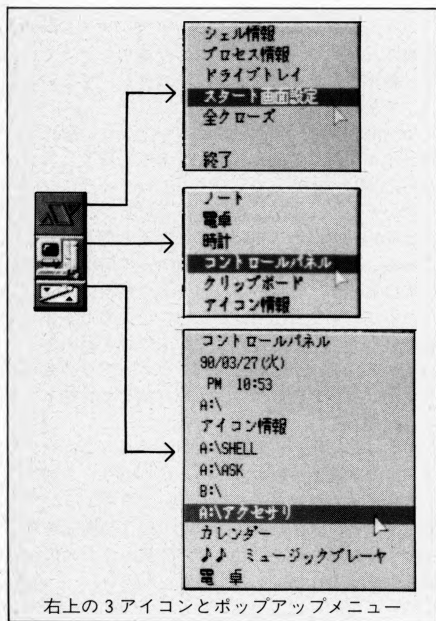
で，お次のページアイコンである。▼をクリックすると２番目のウィンドウがアクティブ（いちばん上）になり，▲をクリックするといちばん下のウィンドウがアクティブになる。また，右クリックでポップアップメニューを出すと現在開いている全ウィンドウが出てくるので，目的のウィンドウを選択すると，そいつがアクティブになる。だから，ウィンドウを開きすぎてあいつはどこに隠れてるんだというイライラに陥ることはないのである。

ついでにもうひとつ。左下にある“X68000”のロゴアイコンだ。これはただのロゴかと思いきや，ちゃんとドラッグすれば移動するのである。そんでもって，ダブルクリックすると，システムドライブのX68000というディレクトリが開くのである。つまりは，ビジュアルシェルでいうQUICKSTARTみたいなもの。てなもんだ。

＊

だいたいにして，テキスト表示が遅い以外はとても面白い。この面白いというのはもちろん褒め言葉だ。ウィンドウシステムなんてのはフレンドリー命だから資源をいっぱい食ってもいいじゃないか。Macintoshなんて，バンドルされてくるマックライトとマックペイントの２本しかソフトがなくてメインメモリが128Kバイトしかなかったのに，OSがフレンドリーだったからユーザーは期待して我慢したのだ（最初はあまり売れなかったけど）。SX-WINDOWにはマックライトとマックペイントにあたるよいアプリケーションはないけれど，その代わりにHuman68kのプログラムはいろいろとあるので，まあ，長い目で見るとしよう。MS-WINDOWSよりは少なくとも，触っていて楽しいのだから。だってフレンドリーでないウィンドウはただのIBMだから（でかくて安心だけど退屈，という意味）。

さて，来月はコマンド/アプリケーション編である。まだまだ**あっと驚く大技**があるので，楽しみにしているように。

▶ところで，先日ついにボックスレンチを買いました。2,800円，結構高かったうえ，フタを開けるとき指を切ってしまいました。もっと角を丸めておけ！
堤 信幸（18）X1turboZ，MSX 大阪府

CHADAWA V

恐怖の読者特集「ちゃだワ５」。今年は２月号で実施したアンケートのメッセージを中心にお贈りしよう。また，日本列島縦断マラソンはハミダシで開催中だ！。

私とパソコンの関係は海よりも深いのだ

◆皆さんはなぜパソコンをやるのでしょうか？特にＸ１やX68000ユーザーで仕事に使ってるという人は少ないでしょう。と，なればン十万円ものローンを組んでまでパソコンに何を求めているのでしょうか？

私がはじめてパソコンに触れたのは２年前でした。私は油の匂いのしない機械が苦手だったのですが，当時，唯一ビデオゲームにハマっておりまして，学校帰りのゲーセン通いはもはや日課でした。そんななか，ふと見つけたのがX68000 ACE発売の広告。「なにっ！　源平ができる！」とゆ〜，かなりミーハーなフザケた理由でなかば衝動買いしてしまったのです。

と，まぁ，ほとんどファミコンと同じノリで買ってはみたのですが，これがかなり違う（当たり前か）なんかいろいろやれるなってな感じで，気づくとゲームよりプログラミングやお絵描きのほうがメインになっちゃってました……。

でも冷静になって考えてみると，こーゆーのってけっこう大がかりなワリに実用性ってあまりないですよね。でも一度パソコンに触れてしまうとパソコンなしの生活って考えられない。以前どこかで「パソコンはなんでもできるが目的がないとなにもできない」とゆ〜話を聞いた覚えがありますが，私はこれからも「なんでもやらせてしまおう」とゆ〜ひどく非実用的な目的を持ってパソコンさんとつきあっていきたいです。

中山　秀隆（19）X68000 PRO-HD，X1C　三重県

◆このパソコンとの間がらももうずいぶんになります。最初は雑誌のプログラムをよく打ち込んでゲームなど楽しんでいました。金欠病で（いまもそうだが）市販のソフトはほとんど買えませんでした。マイコン関係の仕事についているので，パソコンによるプログラムの開発，ハードの設計，自社製品の完成後のチェックにと使う必要が生じてきました。最初は１台だったパソコンもいまは８台くらい，もちろんPC-9801とかPC-286です。会社ではこれらのパソコンで仕事をしています。

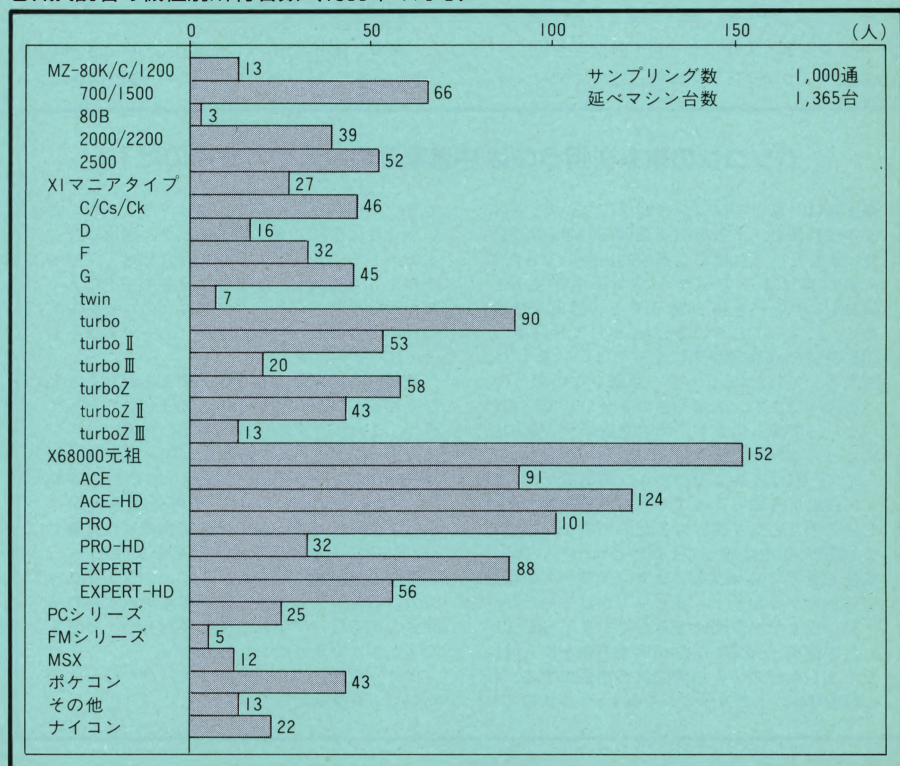
でも家に帰ればわがＸ１turboは8ビットなのです。テレビもOK，パソコン通信もできる，画像の取り込み，FM音源によるミュージック，CG，毎月のいろんなことをプログラムを作って管理したり，外部出力ができるので制御にも使えます（時にはゲームもするけど）。古い機種ですが，私のＸ１は元気です。毎日動いてますので。

松崎　実（36）X1turbo　島根県

中３の時だった。 お父さんが「高校に受かったらパソコンを買ってやる」といって以来，いままで以上にパソコンに興味をもち，どのパソコンがいいか自分なりにいろいろ考え，PC-88VAが欲しいと思い続けていた。で，高校に受かり，ついにVAを買ってもらえるかもしれないと喜んでいたある日，仕事から帰ってきた父の口から出た言葉は「パソコンは店の人に、いまいちばんええのを選んでもらうように頼んどいたからな」だった。これでVAが我が家に来ることはほぼ100％なくなってしまった。そして見せてもらったチラシがX1turboZⅡとX1twinだったわけです。しかたなしにX1turboZⅡにし，しばらく使っていたんですが，これがまた不思議なことにとぉーっても気に入ってしまったわけです。自分としては，ろくにゲームも出ず，近いうちに消滅してしまいそうなこの機種がどうしてこんなに気に入ったのかはわかりませんが……。まぁそれはそれでいいやと深くは考えていませんけど。

横山　博道（17）X1turboZⅡ　岡山県

◆コンピュータ（というよりゲーム）に関心を持ち始めたのは小学１年のころだった。そのころから，ちょくちょくゲームセンターへ行き，親にこっぴどくしかられた思い出もある。ゲームアイデアを考え出す（いま思えばたわいもないものだが）のが最高の楽しみだった。小学４年でPC-6001に惚れ込み，小学５年であのゲームのために生まれたパソコン（？）ファミリーベーシックを買い，BASICをカンペキに（と自分では思う）マスターした。中学１年でMSX2を買って，すばらしいゲームを作ろうとキーボードを乱打したが，できたのはつまらないものばかりだった。技術と反比例にゲ

Oh!X読者の機種別所有者数（1990年４月号）

▲寺門修司（兵庫県）

ームアイデアが減少してしまったのだ。そして中学3年のとき，革命が訪れた。X68000の，言葉ではとうてい言い表せないすばらしさに感動して，親と「公立高校合格条約」を結び，とうとうX68000を手中におさめたのだ。そしていまは「X-BASICは史上最強のBASICだ！」とわめきながら，ゲームを作っている。

木村　啓太郎（16）X68000EXPERT　千葉県

昔（といっても7〜8年前）ぴゅう太という，16ビットで5万円台の完全日本語BASIC搭載のパソコンがあったのを覚えているでしょうか？　私が初めて触れたパソコンは友達の家にあった，そのぴゅう太だったのです。キーを押すと画面に文字が出たことに，とても感動したのを覚えています。初めて手に入れたパソコンが，コモドールのMAX MACHINEというかなりその筋の人でないと知らないマシンでした。恐ろしいことにBASICのフリーエリアが512バイト（Kバイトじゃない），最大でも2Kバイトとメモリが少ないくせに，なんとスプライトを装備していたのです。CPUは6502か何かだと思いました。同じころSC-3000 LEVELII BASICというものがあって，メモリが511バイトで1バイト勝ってるぞ，などと言っていました。そして不可能のないMZ-700を経て，いまは，8ビット最強の（本人は今でもそう思っている）MZ-2500でどこまで行けるか挑戦しています。いつになっても限界が見えてこないんですけどね。

鹿浜　孝宏（19）MZ-700/2500　東京都

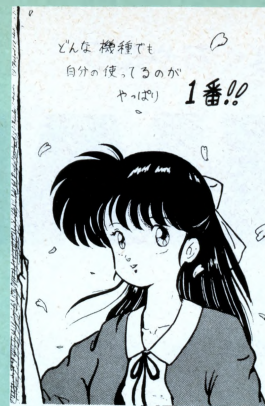
▲富田裕樹（東京都）

▲山崎　浩（広島県）

▲溝畑知幸（兵庫県）

私のまわりのヘンなユーザー

信じられない話

だが，昔，僕の友人Aは「何も書いてないディスクなら，磁性面に触っても大丈夫」と思っていたらしい。彼がいうには「ツメを折っているテープになら，磁石を近づけても問題ない」そうだ。無論，プロテクトシールを貼ったディスクもである。彼は「メーカーもののディスクもすぐにダメになる」といって，ノーブランドのディスクを買うようになった。しばらくして，私は彼に「プロテクトシール貼ったのにデータが消えた」と言われ，家に見にくるように頼まれた。私は，彼が磁性面に触れたためと思っていたけど彼を傷つけたくないので家に寄ってあげた……が，そこで私は笑死した。Oh!Xに知らない人はいないと思いますが，セロテープは，プロテクトシールの代わりにはなりません（色紙をはさめばいいけど……）。

櫻井　貴之（18）X1/turboZ，MZ-80K/C　北海道

◆私のまわりには“その筋”なユーザーがいる。

・Z80の隠れ命令をゾロゾロと書きだす奴。その人に言わせれば，「DD CB d 07」は，「LD A，RLC (IX＋d)」だと言っていた。本当だった。

・PC-8001でレイトレーシングをやった奴。

・ダンプリストを見ただけでどのCPUのマシン語かをずばり見抜いてしまい，またそれを見ただけでディスクアセンブルして書きだす奴。

・雑誌掲載のBASICリストを見ながら頭でコンパイルし，アセンブラで入力する奴。

・「ふん！　画面クリアか，スタックポインタにVRAMの最終アドレスを放りこんで，ゼロをどんどんPUSHすりゃ速いぞ」などと言って，VDPをハングアップさせた奴。

全部本当にいる。

高本　慎一（18）X1/turboII，JR-100　岡山県

◆私のまわりにも変なユーザーが多い。「Cコンパイラってどうやって立ち上げるの？」と聞いてくる奴。

「ハードディスクってどこからディスクいれんの？」という奴。

グロイ（黒い）X68000を持ってる奴。画面に映るダンプリストを見てぶつぶつ言う奴。

怪しい自作ボードのため，前面スイッチで電源が切れなくなり，修理に出した奴。

ま，いろいろあるけど，自分も裸のハードディスクをつないだり，少しならダンプリストが読めたりする。結局みんな怪しい。

渡辺　一矢（20）X68000，MZ-1500　石川県

◆私のまわりにはちょっとどころかとってもヘンというパソコンユーザーが多数いる。なかでも最強なのがS氏である。彼の1日は，ヘヴィ・メタを聴きながら会社へ行くことから始まる。某N○CのV7○などのLSIを設計し，昼休みは「花とゆめ」を読みつつ「ハード・バグが取れない！」と私の会社へ電話してくる。家へ帰ればマンガのポスターと，X68000，MZ-2000，PC-8801FRに囲まれ，ビデオデッキをバラすという，とんでもない人物である。まあ私もS氏もコミケに行っているから一般から見ればちょっとヘンだと思うが……。

小川　真司（22）X68000ACE　東京都

友人Aは

MSXユーザーである。この友人は何かを作るときBASICで組み，「遅いなー，よしマシン語にしよう！」と言ってハンドコンパイルするのである。BASICのリストを見ながらマシンコードを書いていくのだ。ちなみに，この友人はZ80ニーモニックをほとんど知らないのでハンドアセンブルはできないそうだ。

伊藤　直也（19）静岡県

パソコンの未来を担うのは中流家庭の一人娘だったのだ！

◆まずは，私とパソコンの関係について。昭和63年12月某日，ある電気店店頭でX68000ACEと会いました。それはもう運命的出会いでしたね。それ以後はまるでふつうの女子中学生が○GENJIにきゃーきゃー言ってるようにX68000命！　になってしまいました。そしてH.1年10/18。待望のX68000が家にやってまいりました。感激！　でしたね。「もう一生離さないぞっ！」って思ってます。末永くおつき合いしていきたいなーって思ってます。中流家庭の一人娘の私にとってX68000は兄貴ってところかしらね。

さて，私はとある街の小さな電気屋サンに頼んでX68000を取りよせてもらったんです。そこの店の若だんなが届けにきたのですが，彼いわく「僕MS-DOSは知ってんだけどね……」（VSの画面を見て）。さぁ大変。コマンドモードの起動方法がわからない！　リセットかけてみたり，へんなウィンドウ開いてみたりでまいったまいったの連続。その間，Oh!Xが本棚の中を出入りいたしましたが……。1時間試行錯誤のすえ，私がCOMMAND.Xのアイコンを指し，「これなんじゃないの？」と。そこで「やってみよーか」ということになり，ダブルクリック。無事コマンドモードに入り，2人で顔を見合わせて「やったね！」。ちなみに，なぜコマンドモードに入ったかというと，システムディスクのコピーをするためでした。ちゃんちゃん（あー，あほらし……）。

最後に，誰が見ても「こーするんだな」とわかるビジュアルシェルを作ってほしいなーって思う。うん。いまのVS.Xはちょっち使いにくいぞー！　あと，MS-DOSエミュレータあるいは互換性のあるOSを安価で提供していただきたいですね。それと，3.5インチ外付けドライブ。私は文豪を使ってるのでそこで作成した文書もX68000で扱えるとラッキーなんですけど……。

ところで，私はビジュアルインタフェイスってのはパソコンよりも，より多くの人が触れる機会のあるワープロにこそ必要だっ！　と思うんですが，いかがでしょうか。

安井　百合江（15）X68000PRO，びんぼー人のパソコン文豪MINI7HG　愛知県

パソコン界の動向と未来について

◆87年……57.6%，
88年……48.8%，
89年……43.4%。

これ何の数字だかわかりますでしょうか？　実は毎年「言わせてくれなくちゃだワ」に載っている機種別所有数のなかのX1/turboシリーズの構成比なのです。

87年……0%，
88年……18.1%，
89年……29.1%と激増しているX68000とは対照的に減少の道をたどってるのは残念なのですが，まだまだOh!Xの読者のなかではX1/turboシリーズ

が主流だということがよくわかりますね。X68000を買えないX1/turboユーザーのみなさん，せめてこの構成比だけはX68000に抜かれないよう，愛機を手放さないようにしましょうね！
……しかし今年の集計ではすでに逆転されていたりして!?

山田　真裕（20）X1turbo，PC-286VE　神奈川県

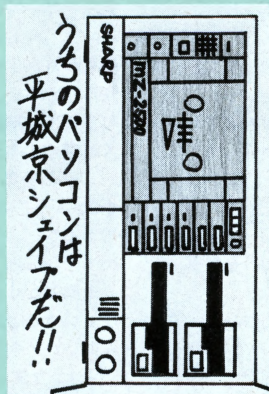
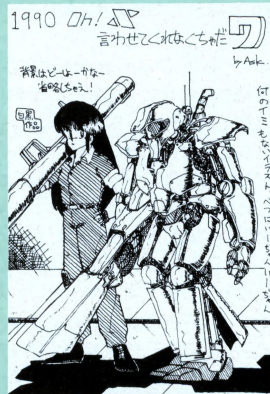
◆これからは，コンピュータというカラを破るようなものが出てくるでしょうね。コンピュータは，もっと人間にとって使いやすくならなければならないと思います。たとえば，SONYが作ったキーボードのないコンピュータ。あれはシャープに作ってもらいたかった。TRONキーボードよりずっとすごいアイデアだと思います。CRTもブラウン管を使うのでは目によくないっ。もっと液晶ディスプレイに力を入れていくべきですね。いや，ディスプレイなどという原始的なものはなくして，眼鏡のようなスコープを使って情報を得るのはどうだろう。ちょうどスピーカーがヘッドホンになるように，ディスプレイはこんなスコープに変わってほしいものだな。ブックコンピュータに使えそうだ。うんうん。

山田　慎也（20）X68000　北海道

TRONチップに興味がある。BTRONにはない。TRONの名前がついているがTRONチップはおとなしい作りの高性能プロセッサに過ぎない。日本もやっと自前のマイクロプロセッサが必要になり，またその必要性ができた。でも，日電のV60/70のように自社製の交換機に使うといった明確な需要がないとリスクが大きすぎる。そこにふってわいたのが坂村健というわけだ。ところで坂本龍馬が薩長同盟を作った本当の目的は倒幕ではなかったという。TRONプロジェクトも別の歴史的評価を受ける日がくるのかもしれない。余談だが，坂村健が切り捨てられたとき，彼は前向きに倒れるのだろうか，後ろ向きに倒れるのだろうか。

鷹屋　光俊（24）X68000，FM-7　神奈川県

◆パソコンを仕事に使うのはキライです。仕事に使うと，趣味でなくなってしまう。だから，遊び心のないパソコンはパソコンとは思わない。たしかに，ビジネス分野で使われないパソコンは販売台数にも限りがありメーカーとしては不利だと思うが，ビジネスはその分野のコンピュータに任せたらよいのではと考える。ビジネスでの活用を考えるあまり，趣味の世界に必要な機能が切り捨てられるのは，いちばんさけてほしいことだ。先のこととはいえ，パソコンが家庭に入ってくるのは明らかなのだから，各メーカーもじっくり腰を落ち着けて，地道にやってほしいものです。

ワープロはビジネス専用とは思えないのですが，Word PROはどうなっているのかな？　標準でワープロソフトを付けた以上（ソフト屋さんも作りにくいですね，一応みんなが持っているから）改良版を出すのはメーカーの責任ですよね？

仲田　富士男（43）X68000 EXPERT-HD　兵庫県

all that's Bug'89

1月号

P.66　マシン語ゲーム工房

X1/turbo用のプログラムで一部説明が抜けていました。PCGのデータはMZ-2500用とまったく同じですので，リスト4にはリスト3の191行以降をくっつけてからアセンブルしてください。

P.72　LAST ONE

このプログラムは実行中のみ，S-OSの一部を拡張しています。ブレイク以外の方法（リセットスイッチなど）で終了しないでください。

P.78　FLICK

キャラクターの一部で標準以外のものが使われていました。以下のように変更してください。

5010_H　A5 A5 A5 A5　→　2E 2E 2E 2E

2月号

P.154　Daddy Mulk

このプログラムをX-BASIC V2.0で実行する際には，あらかじめCONFIG.SYSを，

DEVICE=OPMDRV.X #80

のように設定しておいてください。

3月号

P.38　MZ-700用スペースハリアーをX1で

このプログラムの実行はX1用S-OS“SWORD”から行ってください。

P.96　C調言語講座PRO-68K

i＝1，2，3；という表記は，

i＝（1，2，3）；

の誤りでした。

P.104　FLOAT2＋.X

ファイルサイズが誤っていました。ファイルサイズを整えるプログラムの変数の値を10938にしてください。

がんばれシャープ

◆シャープ，特にテレビ事業部は非常にユーザーに近いメーカーだといえる。シャープはユーザーと一緒になってパソコンの夢を追求しつづけているのである。そうでなければX68000なんていう変なパソコンは生まれないし，サイバースティックなんていうゲームのためだけの周辺機器を作り出すことはできない。100インチの液晶プロジェクターをつなげるなんてのはかなり遊び心が表れていると思う。

蟻田　伸（18）X68000　京都府

◆会社の私の机の上はX68000が半分を占領している。あとの半分を私が使わせてもらっている。ただこれだけであればあまり変わってはいないが，我が社はシャープではなく，某大手電気メーカー。周りのワープロやパソコンはすべて当社製。そのなかでただひとつ異色な光を放つ愛機X68000。会社でゲームをやって遊んでるのかって？　そうではない。一応，仕事に使っているのである。私は物忘れが比較的多いので，私の外部メモリとして使っている。電子手帳PA-8500と組み合わせてスケジュール管理と電話帳，それにBUSINESS PRO-68Kで予算管理や懸案，クレーム管理その他もろもろ。たまにはヘッドホンをして音が周りに聞こえないようにMUSIC PRO-68Kで音楽を聴きながら書類整理も。昼休みはディスプレイはTVに切り替え。なかなか便利。ただひとつクレーム。もっと小さくならんのかね？　ダイナブックでも買おうかな？　早くブック型パソコン開発して！

原　康之（38）X68000 ACE-HD　茨城県

◆ぜひとも作ってもらいたいハードがあります。メモリとキーのみのポケコンみたいなもので，タイプした内容をパソコンなどに入力できるもの。感覚としてはそれ自体で書き換え可能な外部記憶媒体である。RS-232Cを通してパソコンからデータ供給を受けられればなおよい。こんなのがあればいつでもどこでも思いついたことを入力できる。多忙なユーザーのこま切れの時間を有効に使える素晴らしいマシンだと思う。単なるメモとしても使えるだろうし，漢字ROMをオプションにすれば，大したコストもかからないだろう。1万円台で出れば私なら絶対買う。へたなラップトップよりよほど便利だと思う。

北風　裕介（18）X68000ACE，FM77L2　兵庫県

シャープには人情を感じる。

高島　亨（23）X68000北海道

X68000，100万台への野望

◆Oh!Xを読んでX68000を買う決心をした今日このごろ。最近こーゆー人が多いらしいという話を聞いた。つまり，Oh!Xの売れ行きをのばし続ければX68000の売れ行きは倍増し，100万台への野望もすぐに達成できるはずである。というわけでみんなでOh!Xを友人にすすめよう！

伊藤　立治（15）宮城県

X68000が100万台を越えるには，まず「X68000」という言葉自体を有名にしなければならない。TVCMに期待するにはちょっと無理がある。シャープのポリシーからいって難しい。だとすれば，我々ユーザーの力が大きくなってくる。たとえば，ゲームセンターのネームエントリーにはX68000と打ち込んでおく。駅の伝言板に「X68000の前で待つ」と書き込んで去る。手紙を書くとき，意味な

▲鴨居大吾（香川県）

▲鈴木賢吾（北海道）

▲上田孝一（福岡県）

くX68000と書く。友人にビデオのダビングを頼まれたとき，冒頭にX68000のディスプレイ画面を数秒間録画してやる。ソフトハウスにX68000用のソフトを作ってくれとハガキを出す。NECと富士通のカタログを持ち去ってしまう（逆効果？）。Oh!Xをほかの雑誌の上に出しておく。使用機種はなにがなんでもX68000と書く。漫画家にパソコンを描くときはX68000を描いてもらえるように写真を送る。100万人の人にX68000を勧める。これで100万台は達成だ！　すべてはユーザーの力です。

三浦　栄悦（22）X68000PRO　秋田県

◆やはりスーパーリアル麻雀PIIIの移植をお願いしたい。

坂田　務（20）X68000EXPERT-HD，X1turboII，FM-7，MSX2　埼玉県

◆友人がX68000を買う決心をしました。だから「X68000，100万とんで1台への野望」にしましょう。　牧　保志（16）MZ-1500　熊本県

◆X68000，100万台への野望。それはあるジレンマとの戦いであるかもしれない。そりゃもちろん優れたソフトウェアが開発されて環境がよくなるのはいいと思うけど，いまのパソコンの情勢では，台数が増えてもゲームをしたりビジネスにしか使わないような人が増えるだけだと思う。これから100万台に向けては「ユーザーの質」も考えながら進んでいかなければならない。日本のパソコンのなかでパソコンがユーザーの質を問えるものはX68000しかないから，余計にそのことを考えながら増えていかなくてはならない。

アメリカの大学のクラブでは入るために厳しい審査があって，それも身長が170cm以下でないと絶対に入れないなんてのがあるそうだ。そのクラブに入った後もそのクラブの基準に合わなくなると脱会しなければならない。X68000もそういうふうになればちょっと面白いと思う。もちろん，そんなこと現実にはできないけど，少なくともそういう精神をもっていてほしい。しかし，実際にX68000が100万台以上になってユーザーがみんなストロングなんて，考えただけでも楽しく（恐ろしく？）なってしまう。

あ，もしかして「X68000　100万台の野望」ってことは「Oh!X 100万部への野望」ということなのかな？

吉澤　重治（19）X68000，X1F　岡山県

次世代のX68000はこうでなくっちゃ

X68000の新世代はX1に対してX68000のような機械でなければならない。はじめから期待していても，やはり感動してしまうほどの衝撃を持つ機械であることが必要条件である。もしそれが満たされないのならシャープが赤字になろうが開発が断念されようが出す意味はない。X68000のマイナーチェンジで十分である。というわけで，ひとりの頭で作った全体像が正解になるようでも困るが希望としては，

1）　快適なDTPマシンである。

日本人のパーソナルメディアとしてX68000をみた場合，DTPマシンであるというのは必要だといえる。ワープロが急速な進歩を遂げたとはいえ欧米のタイプライタに比べ「快適」のレベルに達したとはいえない。筆記用具を駆逐するほどの機械であることを期待する。

2）　買ったときからネットにつながる。

はじめから分散処理型ネットにつながる。ソフトウェア資産の開発，活用，発展の面でネットワークは不可欠。シャープが独自にデジタル回線を全国に作るのは……無理かなぁ。

これだけの条件を備えていればいくら高くても文句はいわない。電光石火で広まりすぐに量産で安くできるだろう。

田辺　浩靖（19）X68000PRO，X1/turbo　千葉県

◆X68000の方向のまま，次世代マシンを考えると，カラー版NeXTになってしまうような気がします。そこそこ安ければそれでも十分に売れるでしょうが，まったく別の方向も期待したいところです。たとえば，

1）　アメリカのザイリンクスというメーカーは，ユーザーがプログラムできるゲートアレイを作っています。これを台帳にのせておくことで，RISCをはるかに越える専用ハードウェアをソフトウェアごとに作りなおすことも不可能ではありません。DSPだのRISCだの，遅くて待ってられないほど速いマシンができるはずです。

2）　いまアイデアをためている段階なので，どのようにしたらよいか具体的に不明なのですが，

▲伊藤大地（東京都）

▲小井田伸雄（岩手県）

▲小林貴洋（千葉県）

▲住友智代（愛媛県）

今月はSTUDIO Xは「言わせてくれなくちゃだワ」に吸収されてしまいました。と，いうわけで「言わせて内臨時出張STDUIO X」を開設。3月号にハガキを送ってきてくれた人，ここに大集合!!

◆「なんでも鳴らせるOPMD.X」を作ってくださった西川善司様は神様です。いままでOPMAを使っていましたが，自分でサンプリングデータを入力しようとすると，X-BASICでリズムのデータをシークして（これが大変！）頭出しをしたうえで自分のデータを上書きしなければいけなかったのです。ずっと不便だなぁ〜，と思っていました。でもOPMDはコンフィギュレーションファイルによってそれをあっさりと解決し，どんどんデータが追加できるのがすごい。またMIDI対応の部分も，将来MIDI楽器を買ったとき使えそうです。今日は&H1679番地まで打ち込んで疲れました。明日にはOPMDが使えるでしょう。楽しみだなぁ〜，とっても！

山口　隆久（17）　X68000EXPERT-HD　東京都

◆OPMDはOPMAよりもさらに素晴らしい。しかし最近思ったのだが，ドラムスはFM音源で無理すれば鳴らせるが，私がいちばん使いたい女性コーラスがFM音源ではとうてい無理であるということだ。女性の「アァ〜」というコーラスがもし自分の曲に使えたらどんなにいいだろう。

五島　智明（18）　X68000　長野県

◆特集の「MMLを楽譜データに」での「.SCOへの変換」は，涙うるうるものでした。次回作品としてポルタメントやソフトLFOをサポートしたOPMD上位コンパチのOPM?ドライバを期待しています。　　岡田　隆裕（18）　X68000　埼玉県

◆読者の皆さん，ベートーベンのピアノソナタ「ハンマークラヴィーア」って知ってますか？　私は中2のころ「もっとも難しい」と書かれているのを見て練習を始め，高校生になって第1楽章が弾けるようになりました。でも，モノにはできず，「あとの楽章はX68000に弾かせてやる！」なんて思ったり……。

江原　忠士（19）　MZ-2000　岡山県

◆私のX68000のディスプレイはCZ-604Dですが，後ろからコードを引っ張ってきてスピーカにつなげるのには驚いた。妹は「さる耳テレビ」といって馬鹿にするし。CZ-604Dユーザーよ，怒れ！

奥津　明彦（21）　X68000PRO　宮城県

◆DōGAのアマチュアCGコンテスト発表会に行ってきました。少し早めに着いたら席が空いていたのでどうなることかと思いましたが，開会直前には満席となりました。作品では「デファイナブルファンクション」が好きでしたが，制作の内輪話を聞いたときは，ただただ頭が下がるのみでした。VTRのコマ撮りをしていてデッキが1台昇天したそうです。さーて，来年が楽しみだなっと。

田中　義彦（26）　X1C　東京都

◆DōGAの上映会を見ました。が，X68000ユー

▲笹川明大（徳島県）

all that's Bug'89

4月号

P.44 OPMA.X

リスト５に誤りがありました。

```
30 n1 =fopen("opma.$$$","r")
40 n2 =fopen("opma.x","c")
```

のように訂正してください。

P.59 Like The Wind

1580行右端の“5b”の後ろに“：”が抜けていました。追加してください。

P.131 System-7B

9008_H HANTEIルーチン解説部の戻り値が誤っています。

　Cy=1：接触している
　Cy=0：接触していない

に変更してください。

$9A9B_H$ STTM，$9AFF_H$ MVTMの戻り値でIXレジスタとIYレジスタが入れ替わっていました。STTM側をIX，MVTM側をIYにしてください。

COLORMASK@のアドレスが誤っていました。

　$989A_H$ → $989F_H$

にしてください。

また，起動時に画面がクリアされていないと画面にゴミが出ていましたが，これは$9FC0_H$と$9FDD_H$を09_Hにすることで直ります。

$9E9B_H$ PRINTMENUXのパラメータ部はIX+2とIX+3に関する内容を入れ替えたうえで，以下のものを追加してください。

　ＨＬ′＝転送元仮ATRのアドレス
　ＤＥ′＝転送先仮ATRのアドレス
　Ｂ′　＝横の長さ
　Ｃ′　＝縦の長さ
　ＨＬ　＝003EH（画面転送ルーチンの場所）
　（IX＋13）のビット０が１ならほかの転送ルーチンを使用。

P.133 System-7B

TRANS40 エントリアドレスが$9256A_H$ となっていました。正くは9256_Hです。

SMPOUTを使用する際にはあらかじめ$E004_H$に周波数（通常３）に設定しておく必要があります。JUMPで「PCとDEを加える」という解説がありますが，正くは「PCにBCを加える」となります。

また，PRESSの戻り値が抜けてしまっていました。HLレジスタに圧縮されたデータの最終アドレスが格納されています。これは，場合によってはもとのデータより大きくなることがあるので注意してください。PRINT，MESSAGEなどでは，CHR$(13)がエンドコードとなります。割り算ルーチン（$90D0_H$）の戻り値で商と余りが逆になっていました。また，このときの被除数(HL)の範囲は０～$FFFF_H$，除数(Ｅ)の範囲は１～FF_Hです。なお，起動は必ずROMモニタから行ってください。

5月号

P.66 X-BASICでMIDIコントロール

リスト11 MD－OUT.Sの12行目に誤りがありました。次のように修正してください。

```
move.l $40(sp), d1
```

P.124 戦略的ライトサイクルゲーム

X1版で座標系の変換を間違えていました。

```
1190 fda=POINT(X, Y)
```

に修正してください。

P.154 RING

SOURCERYからの変更プログラムはデバッグの状況を問わず有効ですが，作成されるオブジェクトコードが掲載されているものと異なる場合があります。動作には問題ありません。

人間の精神構造を組み込んだマシンにしてもらいたいと思います。仕事でこれを実現する予定なので期待していてください。

　清水　雅夫（30）X68000　神奈川県

◆UNIXはやめてくれーい。ぜ～ったいだよ。あんな，でっけぇOSのせて，パーソナルワークステーションでもあるめぇ。CPU？　そんなもん関係あるかい。本来CPUは影の力持ち。素晴らしいソフトが作れるのなら，素晴らしいソフトを走らせることができるのなら，なんだっていい。シャープオリジナルでも。それより，周りのLSIに全力投球すべし！　特にX68000の代名詞であるグラフィックと音に関しては，決していいかげんではならぬ！　でも，もうおおかた概略も決まってしまっていてどうにもならないんでしょうねぇ。

　梶川　達也（25）X68000ACE-HD，S1　岡山県

◆とにかく真のハイパーメディアパソコンを目指してほしい。ハイパーメディアというのは，あらゆる情報が同時に扱えることをいう。つまり，エディタでプログラムなどを作成中に，ウィンドウを開き，テレビを映す。また，別のウィンドウを開き，そこに前のテレビ画像をCOPYできたり，プログラムのコメント部分に音声データを置いてマウスでクリックすると音声でプログラムを説明するなど，とにかく目と耳で感じられる情報のすべてを同時に扱わなくてはいけない。最低でも「Macのハイパーカードに VTRデータを扱えるようにした」ぐらいの機能は必要だ。次世代のX68000はそのハイパーメディアを目指して作ってほしい。

柴田 俊（19）X68000PRO，PC-8801MR　神奈川県

私も初心者なので，えらそーなことはいえないが，ひと言いわせてもらいます。マシンが進化するのは，我々ユーザーにとって喜ばしいことであるけれど，X68000を超えるマシンが現れたところで，果たして使いこなすことができるかということが非常に疑問です。これ以上進化すると，ブラックボックス化してしまうのではないでしょうか。私自身X68000というマシンの，まさに氷山の一角しか使いこなしていません。もちろん，私だってX68000の次世代マシンには興味があります。夢見ることは必要だけれど，それには私たちユーザーがもっともっと進化しなければならないのではないでしょうか。

今井田　和也（17）X68000ACE-HD，X1G　愛知県

◆X68000の次世代マシンは，やっぱりゲーセン機を超える画像処理能力と，ミニコンにもせまる処理スピードということで，CPUは68030を４つ，画像関係は３Ｄスプライトを装備。これは立体の物体を定義でき，そして回転もHEAD，BANK，PITCHの設定ができ，位置決めも，Ｘ，Ｙ，Ｚ軸でできるというもので，それに割り当てられるメモリも1Gで，物体の大きさも，８×８×８からメモリの許す限り大きくすることができるというすぐれものである。そして，グラフィックは1024×1024のザーはまだマシンの力を持て余しているみたい。表現を欲張ってストーリーまでは力が入ってないような気がするけど。自戒，自戒。

臼渕　啓明(23)　X68000，X1turbo，MZ-80K/2000　神奈川県

◆２月19日の朝日新聞の夕刊に島田雅彦氏の「見える壁，見えない壁」という評論が載りましたが，その日読んだ「お茶目な計算機たち」とあまりにも内容がシンクロしているのでビックリ。有田氏がノスタルジアを「病」としているのに対して，島田氏は「見えない壁」としているのです。いまの状況を同じように読み取る人はいるんだなと感心しました。　石川　孝子（26）　なし　愛知県

◆「(で) のショートプロぱーてぃ」に投稿するため日夜アイデアを考えています。そこで考えたのが超能力を開発するゲーム。ディスプレイに隠れている☆△□○を当てるという簡単なものです。しかし，これだけではアイデアに欠けると思うので，もうちょっと味をつけるため悩んでいます。

　遠藤　亮司（20）　X1turbo　栃木県

◆（で）氏へ質問。アイデアだけはかなりのところまでいきながら，ある事情でゲームにできない作品があるのですが，そーゆーものでも投稿していいのでしょうか？（さてどんな事情だろう，フフフ……）　金丸　勉（18）　MZ-700　滋賀県

◆Oh!X LIVE in '90が好きです♡

　森　夕香（13）　X68000ACE-HD　滋賀県

◆「X-BASIC調理実習」の「ギターで遊ぼう」は面白かったぜ。マウスでギターを弾くなんてかっこいーじゃん！　ところで私，ストラトタイプのＥギターを１本（フェルナンデス）持ってます。MT-32とシンクロさせてみたいんですが。

　浅利　拓志（22）　X68000，X1C/turbo　宮城県

◆「X68000マシン語プログラミング」の単行本化はぜひお願いします。Oh!Xを何冊も開いて勉強するのがきつい……。さらに最近は○月号参照というのが回を重ねるごとに多くなってきたので。１冊にまとまるといいです。

　若林　英生（25）　X68000　神奈川県

◆「ペンギン情報コーナー」が見開きだととても見やすい。それはそうと今回のIOデータのRAMボードのようにPC-9801用で有名なメーカーがX68000用の製品を販売してくれるのはとてもうれしい。でも，ヒネくれて考えると古巣に帰ってきたとも考えられるけど……。

　中内　崇夫（21）　X1turbo　神奈川県

◆……あの，３月号の「ペンギン情報コーナー」のマッサージ棒のイボイボが目に焼きついて，焼きついて……。

迫田　賢一(39)　X68000，X1，MZ-2000　大阪府

◆３月号の「Oh!X readers'ぎゃらりぃ」のコーナーで，スタッフの高橋哲史氏の年賀状は，なぜ「うし」なのでしょうか？　1）　干支を知らない，2）馬のつもりで描いた，3）　氏にとっては「うし」年である。

中野　義則（22）　X68000PRO，X1turboIII　新潟県

◆Oh!Xで「イカすプログラム天国」といった感じで，アマチュアプログラムのコンテストをしたらどうでしょうか？　毎月送られてくるプログラムを編集者の独断と偏見で選び，グランドチャンピオンを選出するものです。

　藤山　健二（18）　X68000，MZ-1500　愛知県

all that's Bug'89

6月号

P.61　学習リモコンの製作

サンプルプログラムをアセンブルする際にはCコンパイラに付属するDOSCALL.MACとFDEF.Hというファイルが必要です。Cコンパイラをお持ちでない方は，リストAのファイルを作成してFDEF.Hとして使用してください。DOSCALL.MAC については「X68000 マシン語プログラミング」の場合と同様，Human68kユーザーズマニュアルを参照してファイルを作成してください。

P.161　質問箱

X1turboでBIOS内ルーチンを利用する際に割り込み禁止の指示が抜けていました。BIOSを呼び出すときは割り込みを禁止したうえで，8000_H以降に処理ルーチン，スタックを置くようにします。

リストA

```
 1:   nlist
 2: *
 3: * fdef.h X68k XC Compiler v1.01
 4: * Copyright 1987 SHARP/Hudson
 5: *
 6: *  引数コード  dc.w  ????
 7: *
 8: float_val   equ  $0001   float型の値
 9: int_val     equ  $0002     int型の値
10: char_val    equ  $0004    char型の値
11: str_val     equ  $0008     str型の値
12: *
13: float_omt   equ  $0081   省略可能なfloat型の値
14: int_omt     equ  $0082   省略可能な  int型の値
15: char_omt    equ  $0084   省略可能な char型の値
16: str_omt     equ  $0088   省略可能な  str型の値
17: *
18: float_vp    equ  $0011   float型の変数の値のポインタ
19: int_vp      equ  $0012     int型の変数の値のポインタ
20: char_vp     equ  $0014    char型の変数の値のポインタ
21: str_vp      equ  $0018     str型の変数の値のポインタ
22: *
23: ary1        equ  $003f   1次元配列（全ての型）
24: ary1_i      equ  $0032   1次元配列（int型）
25: ary1_fic    equ  $0037   1次元配列（float,int,char型）
26: ary1_c      equ  $0034   1次元配列（char型）
27: ary2_c      equ  $0054   2次元配列（char型）
28: *
29: float_ret   equ  $8000   返り値はfloat型
30: int_ret     equ  $8001   返り値はint型
31: str_ret     equ  $8003   返り値はstr型
32: void_ret    equ  $ffff   返り値はなし
33: *
34: *  引数オフセット  sp+????
35: *
36: par1     equ   6   第1引数FAC
37: par2     equ  16   第2引数FAC
38: par3     equ  26   第3引数FAC
39: par4     equ  36   第4引数FAC
40: par5     equ  46   第5引数FAC
41: par6     equ  56   第6引数FAC
42: par7     equ  66   第7引数FAC
43: par8     equ  76   第8引数FAC
44: par9     equ  86   第9引数FAC
45: *
46:   list
```

RGB各8ビットで1677万色，テキストも同様 メインメモリは4Gで……。

書いててこわくなってきたので自爆!!

菅原　尚伸（19）X68000EXPERT，MZ-2000　岩手県

ソフトハウスさんにもお願い！

◆なんだってあんなに高いソフトしか作れないのだろう？　電脳倶楽部で祝氏がいっていたように，「レンタル業者がいなくなれば，プロテクトをかける必要がなくなって，価格を今の半分にすることができる」というのは明らかに“ウソ”である。まったくハラだたしい！　ソフトのマニュアルを豪華にしたり，パッケージをやたら大きくしたり，そういう行動をへらしてでも価格低下にはげむべきである！　もっとも，悲しいことに自分を含めて最近では価格をみてソフトの内容を判断してしまう傾向がある。安いソフトが出にくいのもわかるような気もする……。

伊東　新仁郎（22）　X68000EXPERT　北海道

◆これは何度も言っていることですが，ソフトの値段が高い！　いくらバイトをしているボクでもこれはつらいです。レンタルがなくなったというのに，ゲームは値下がりするどころか高くなるばかり。ソフトのレンタルはコピーされてしまうから認められないと聞きましたが，パッケージもないし，マニュアルも，ユーザー登録さえできないんだからいいんじゃないかと思いますけど。

とにかく早くソフトを安くするか，レンタルを認めるか何かしないと，みんなパソコンゲームから離れていってしまいますよ。ぼくもレンタルがなくなってからゲームをやる量が減りました。しょうもないソフトを買ってしまったときのショックは口では言い表せません。ホントにどうにかしてください！

野瀬　茂樹（20）X68000ACE　大阪府

▲伊藤圭一（埼玉県）

4096色でるし，FM音源も8chもある。なぜ，ソフトハウスはこれに気づかないのだ！　もっとX1ソフトを出せー！　くそー！　ほかの機種がうらやまし～～!!

仲本　英生（15）　X1turboZ，MZ-700　東京都

◆とにかくゲームのタイトルなどに英語（アルファベット）を使うのはやめてほしい。まぁゲームの舞台が英語圏の国ならまだしも，地球から遠く離れた宇宙戦争や，別の世界の光と闇の戦いに英語（アルファベット）が出てくるとゲーム制作者の想像力の乏しさを感じてしまう。英語のほうがかっこよく聞こえるという安易な理由でゲームの顔であるタイトルをつけるのは手抜き以外のなにものでもない。せめてゲームに合った言語を選んでほしい。

これは別にゲームに限ったことではないのだが（職安がハローワークだと？　なに考えてんだ！　なにがWinkだっ！）。とにかく日本語でも十分に詩的で魅力的なタイトルは作れるのだし，なるべく日本語を使ってほしい。私がゲームのなかでいちばん好きなタイトルは“38万キロの虚空”だ。

小谷　恒（19）X1turboIII　岡山県

とうとう第5回目を数えることになった恐怖の読者特集「言わせてくれなくちゃだワ」。今年は2月号で実施したアンケートから採用した皆様のメッセージを中心に構成してみました。なお，日本列島縦断マラソンはハミダシで開催しております。今年も元気にいってみよー！

◆私は「X68000が欲しい，欲しい」と朝から晩まで考えている学生です。しかし恥ずかしながら最近までOh!Xの存在は知りませんでした（Oh!PCやOh!FMは知っていた）。そんなわけで今月でまだ2冊目ですが，長いおつき合いになると思いますのでよろしくお願いいたします。

藤田　清孝（21）　なし　東京都

◆X68000を買おうと決めたあの日から……，CPUは68系でなければダメだと思ったあの日から……，某マシンに失望したあの日から……，FM-7ユーザーだった僕とOh!Xとの関係が始まった。Oh!Xを買い始めて満1年。これからも買い続けるだろう。

久下沼　信（20）　X68000ACE-HD　石川県

◆僕はなんてバカなんだ。Oh!Xを3月号から定期購読にしたのにそれを忘れて3月号を買ってしまった。　山崎　幹生（15）　X68000ACE　新潟県

◆ついに「マイコン BASIC Magazine」がOh!Xと同じ大きさで，Oh!Xより厚くて，Oh!Xより安くなりました。本棚のスペースがまた減ってしまうよ～。でも僕はOh!Xの味方です，Oh!Xだけは絶対に不自由させないように本棚に入れてあげるからね！うん。　高橋　智宏（17）　X68000　愛知県

◆ソニーのMDp-111とパナソニックのれんたろうを買いました。冬休みのバイトの成果です。これで私もOVAが見られます。私は高河ゆんが好きです。　鈴木　賢吾（19）　MZ-2500　北海道

◆1泊2日でスキーに行って帰ってきたら部屋中水びたしだった。ボーゼンと立ちすくむ僕。なんと上の階の人の水もれが原因だった。シャワーを完全に止めなかったのが夜になって凍りつき，下水に流れなくなったそーです。損害は弁償してくれるからいいけど，3年分のOh!Xが水びたしなのは弁償のしようがない。せっかくの貴重な資料が……。寒冷地に住んでいる人は水の扱いには十分注意しましょうね。損害賠償としてOh!X1冊について1万円取ってやろーかな。

安永　吉徳（20）　PC-9801UV11　長野県

◆僕はやっと高校に合格できて入学祝いとしてX68000を買ってもらうんですけど，全然わからないんです。ハードディスクってなんですか？MIDIってなんですか？　通信をするには最低限なにが必要なんですか？　そういうこれから始める人のための基本的なことを知りたいんですが。

酒井　智志（15）　なし　東京都

◆昔，X1turboとPC-8801SRとどちらを買うかずいぶん迷いましたが，X1を選んで正解だったと思っています。そしていま，私はなんの迷いもなくX68000を選びました。X68000はみんなの夢がつまったマシンだと思っています。

堀本　直宏（20）　X68000，EXPERT-HD，X1turbo　奈良県

◆やっとX68000　EXPERTを入手し，最近ようやくX-BASICがわかってきたところです。そうしてそのスピードの限界も……。で，ついにCコンパイラを買ってしまいました。というわけでCやアセンブラの勉強中です。そのうち投稿することもあるかもしれませんのでよろしく。でも，X68000はなんかプログラムを組まなくちゃいけないと思わせる不思議なパソコンですね。

長谷川　亮（19）　X68000EXPERT　神奈川県

◆X68000の新製品が出るときに備えて春休みにアルバイトをしようと思っていたのだが，私の住んでいるところは田舎なので仕事がない!!　計算

ちょっと身近な面白い話

◆というわけで私はドラクエⅣを買った。発売後まだ１週間というのにここ台北でも売っているではないか。15,000円は少し高かったが，店員いわく「日本ではほかのソフトも買わなければ売ってくれない」まっ，よくご存じですこと。ちなみにハードは安い。6，000円くらいで買えます。メガドラもIBM-AT/CTも安い。ぜひ一度遊びにいらっしゃい。　　　　高綱　慎二（32）台湾省

◆先日，我が家のネコがX68000が置いてある机にのぼって遊んでいた。私がふと見ると“マウス”を動かしていた。そのとき，うちのネコは３日間ごはんをやっていなかった。マウスが喰われるぅーと思ったが，無事でよかったよかった。

清水　拓詞（18）X68000ACE　北海道

◆X68000ACE用の1Mバイト増設RAM（CZ-6BE1A）を買いに行ったらどこも品切れ。だが，某店に入ってみると３つあった。価格は税込みで￥33,166。店員との交渉も虚しく，どーしても欲しかった私は結局そのままの値段で買ってしまった。店を出て，家路につこうとしたとき，私は友人に教えてもらった店を思い出し，行ってみた。私の目に入ったものはCZ-6BE1Aであった。おそるおそる店員に聞く。

私「あのー，このCZ-6BE1Aはいくらですか？」
店「えーと，定価はいくらやった？」
私「確か￥38,000だったと思いますが？」
店「じゃあ￥30,000や。」
私「そ，そーですか……」

次の瞬間，私は高速でさっきの店に戻り，「あのー，誠に申し訳ないんですが，急に現金が必要になりま……」といかにも残念といった表情をつくって嘘八百を並べたて，CZ-6BE1Aを返品し，またもや高速で例の店へ行って購入しなおしたのであった。フハハハッ！　私は日本橋に来ると，神にも平気で屁をかます男となるのだっ！

阿部　裕司（21）X68000ACE，X1，MZ-700　大阪府

浪人生への道

第三話

80回目のコンティニュー。またティキは生き返ってクック山を登っていった。

all that's Bug'89

7月号

P.32　透視変換アルゴリズム

「左手系」と「右手系」の用語の使い方が逆になっていました。正しくは論理座標系が「右手系」で，デバイス座標系が「左手系」です。

また，透視変換を行う際の計算式のままだと場合によっては面の前後関係が逆になってしまうことがありました。それを防ぐために「正規透視座標系」という座標系を用います。35ページに書かれていた，

$$Z_5 = d \cdot Z_4 / (d - Z_4)$$

の計算式を，

$$Z_5 = 1.0 / (d - Z_4)$$

に変更してください。なお，この変更に伴い，７月号に掲載されたリストの17，18，591～594行も変更したほうがいいようです。８月号に掲載されたリストと見比べて訂正してください。

8月号

P.154　CP/M用ファイルコンバータ

CP/Mには最初のセクタを０と数えるか１と数えるかで２種類が存在します。以下はセクタ０オリジンの機種との共用化変更点です。共通部分を，

```
0146  C3 2C 0B 00 00 00
0341  CD 90 0B 00 00 00
03B6  CD 90 0B 00 00 00
04B8  65
0706  C3 81 0B
```

に変更後リストＢを追加し，起動時に０または１の各機種で適したセクタオリジンを指定してください。また，SLOADコマンドで/Ｔオプションをつけるとファイル内容を画面に出力できます。

リストＢ　0B2C～拡張部分

```
ADRS +0 +1 +2 +3 +4 +5 +6 +7 Sum
0B00 00 00 00 00 00 00 00 00:00
0B08 00 00 00 00 00 00 00 00:00
0B10 00 00 00 00 00 00 00 00:00
0B18 00 00 00 00 00 00 00 00:00
0B20 00 00 00 00 00 00 00 00:00
0B28 00 00 00 00 5E 36 00 23:B7
0B30 56 36 00 23 ED 53 9D 0B:97
0B38 11 08 00 19 7E 23 66 6F:A8
0B40 7E 23 66 6F 22 F1 02 11:9C
0B48 5D 0B CD FF 01 CD 0E 02:12
0B50 D6 30 FE 02 30 F1 32 9C:F5
0B58 0B CD EA 01 C9 49 6E 70:B3
0B60 75 74 20 46 69 72 73 74:11
0B68 20 53 65 63 74 6F 72 20:B0
0B70 4E 75 6D 62 65 72 20 28:B1
0B78 20 30 20 6F 72 20 31 29:CB
--------------------------------
SUM: 26 D5 2D 27 99 17 E9 A1 64B4
```

```
ADRS +0 +1 +2 +3 +4 +5 +6 +7 Sum
0B80 00 0E 01 CD 4C 01 ED 5B:71
0B88 9D 0B 73 23 72 C3 00 00:73
0B90 2A 9C 0B 26 00 19 22 FF:31
0B98 03 44 4D C9 00 00 00 00:5D
0BA0 00 00 00 00 00 00 00 00:00
0BA8 00 00 00 00 5E 36 00 23:B7
0BB0 56 36 00 23 ED 53 9D 0B:97
0BB8 11 08 00 19 7E 23 66 6F:A8
0BC0 7E 23 66 6F 22 F1 02 11:9C
0BC8 5D 0B CD FF 01 CD 0E 02:12
0BD0 D6 30 FE 02 30 F1 32 9C:F5
0BD8 0B CD EA 01 C9 49 6E 70:B3
0BE0 75 74 20 46 69 72 73 74:11
0BE8 20 53 65 63 74 6F 72 20:B0
0BF0 4E 75 6D 62 65 72 20 28:B1
0BF8 20 30 20 6F 72 20 31 29:CB
--------------------------------
SUM: F0 CE F9 06 57 F4 F8 FB F5D5
```

▲神田倫明（東京都）

▲中島敦夫（岡山県）

▲清水健太郎（静岡県）

が狂ってしまった。

泉　昭彦（19）　X1turbo　三重県

◆諸行無常じゃのう。そして，時代はX1からX68000へと移っていく。さらに，驕る日電久しからずなのである。ところが，盛者必衰だったりするのでX68000も気をつけましょう。

渡邊　裕之（17）　X1turboZⅡ　北海道

◆いまパソコン界は長い冬ともいえる冷たい時代を迎えている。X68000，FM-TOWNS，PC-9801などのユーザー間でかつてないようないがみ合いが続いているからです。でもパソコンは使ってこそ意味があるもの。たとえMZ-700でもスペハリはできるのです。無駄な争いはやめて精進しましょう。

阿部　文明（17）　X1turbo/Ⅱ，X1F，MZ-700　千葉県

◆X68000やっててホントによかった!!　現在初代X68000に１Ｍバイト増設，トランスピュータ，コプロセッサ，MIDIボードを入れ替えて使い，イメージユニット，ハンディスキャナ，MT-32，カシオのサンプリングボード，モニタアンプ，サイバースティックをつなぎX1turboⅢも周辺機器化している。なにが言いたいかというと周辺機器に金をかけたくなるマシンはX68000だけだということです。

山川　秀幸（21）X68000，X1turboⅢ　千葉県

◆パソコンでなにか変わったことをやろうとすると最初にお金の問題にぶつかる。しょせん，パソコンはお金持ちの道楽なのかもしれない。パソコンが日常生活の一部となるのはいったいいつの日だろうか？

泉　哲也（19）　X68000ACE，MZ-700　岩手県

◆ついにX1turboZⅢを買うことができました。しかし，最近は中学生がX68000を持っていても珍しくない時代ですからねぇ，しみじみ（うっ，じじくさくなってしまった）。

山野辺　太郎（14）　X1turboZⅢ，MZ-100/2000　宮城県

◆おおおおーっ，今日（2/20）地震がありました。ドキドキしながら部屋へ行くとマンハッタンシェイプのX68000はビクともしておらず，ディスプレイのほうが台から落ちかけていました。うーん，さすが，高層建築の耐震構造。

大野　二郎（23）　X68000ACE-HD，パソピア７　静岡県

◆Oh!Xの広告でもご存じの計測技研の増設メモリ＋コプロセッサボードですが，コプロセッサなしで購入し，MC68881を装着し，サンプルプログラムを実行してみた。デバイスドライバはFLOAT3＋.Xなので，さすがに速い。しかしMC68882に替えたらまったく動かない。たぶん周辺のデバイスとのタイミングが違うのだろう。

今井　喜久夫（30）　X68000ACE-HD　東京都

◆しかし，シャープは一流のワープロメーカーであるのに，そのノウハウをそそぎ込んだワープロソフトをX68000用に出さないのでしょうかね？「書院　PRO-68K」なんか出したら売れると思うんですけど。

櫻井　良多郎（18）　X1G/turboZⅡ，X1G　東京都

◆X68000のマシン語でゲームを作ろうと思いました。しかし，いざ作ろうとしても参考となる資料がなくて，作るに作れず……。というわけで，ゲーム作りに必要なことを載せてください。

佐村　和亮（16）　X68000EXPERT　山口県

「えーい，死ね死ね死ね死ね————い」そして
ピューピューとの再会。涙のエンディング。
「さあ，やっと勉強でける。えーと，はぁtanθの不定積分？　こんなんでけんの？　げーん，わからん。物理にしよ。コンプトン効果？　なんやそれ？えーい化学じゃ。フェリシアン化カリウム？……」
そして景清は頼朝に最後の闘いを挑んだ。命をかけて……。
「えーい，死ね死ね死ね死ね————い。」

佐藤　能久（18）X68000，PC-6001mk2　大阪府

◆1988年の夏のある夜，祖父宅にいた私は夕食を食べていた。と，伯父が仕事から帰ってきた。手にはダンボール箱。“いいもの持ってきた”と取りだした缶には黄色いロゴで「麦コーラ」（いつのころからか「大麦健康飲料」にかわったが）。「高級品だ。こんど韓国から輸入することになったんだ」……これが，あの，いまやOh!Xではメジャーになったメッコールとの出会いであった。それ以後，家にはメッコールが常備されていて，伯父と，それ以上に高2のいとこが「こんなウマいものはない」と愛飲している。私はといえばそれ以後飲んでいない。
ところで，このメッコールの意外な利用法。これが煮豚のタレになるのである。なべに豚肉のかたまりを入れて煮るだけ。他になにも調味料を使わなくてもスバラシくおいしい煮豚になるのである。メッコールに弱い人でも，一口食べれば“おほぉいしひいいっ”っということマチガイなしである。伯母が作ってくれたのだが，“タレはメッコール”と言われたのにも関わらず，すべてたいらげてしまった。
ちなみに，先日伯父に“雑誌にメッコールのことが載ってるよ”と言ったら，伯父の反応は“まずいって書いてあるのか……”であった。チャンチャン。　越川　直樹（20）MZ-2500　群馬県

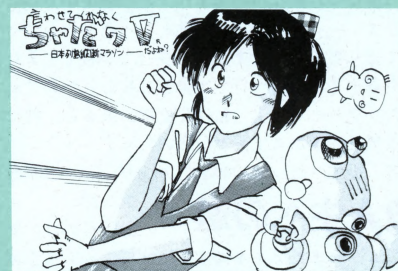
——ビデオ配布のお知らせ——

不幸は再び……

プロジェクトチーム DōGA　かまた ゆたか

先月のカラーページで第２回アマチュアCGAコンテストの入選作品の発表を行いましたが，作品をご覧になっていないとわからない話も多くありました。しかし，ご安心ください。当チームでは，全国100万人(?)のOh!X読者のために，入選作品を集めたビデオテープを配布することにしました。
ビデオテープ配布を希望する声はかなり以前からありました。それにもかかわらず，連載中その話題を避け続けていた理由は２つあります。
ひとつめの理由は実費が高くなりすぎることです。まずビデオテープ自体がCGAシステムのようなディスクよりずっと高価だし，郵送料もかさみます。さらにダビングは専門の業者に依頼する必要があるため実費は2000円ぐらいになります。CGAシステムの実費が3000円なのを考えると，どうしても割高感は拭いきれません。
もうひとつの理由は当チームの手間の問題です。ダビングや発送の作業は業者に依頼できるとしても受け付けはDōGAが行わなければならず，そしてトラブルが……。CGAシステムの配布で起きたあの惨劇が再現するかと思うと，スタッフ一同思わずしり込みしてしまったのでした。そして都合の悪いことはすっと忘れるという特技を持つ当スタッフは，ビデオ配布のことなど忘れて，いたって健康的な生活を送っていたのでした。が，そこに，平和を打ち砕く不幸の電話が……。
K「こんにちわ～。某Ａ編集部の某Ｋです。CGAコンテストのビデオ見ましたけどホント素晴らしい作品ばかりですね。うちでもカラーでとりあげることになりましたので」
かまた(以下か)「どうもありがとうございます」
Ｋ「それでですね。やっぱり写真だけじゃあのスゴさはわからないと思んですよ。読者からもビデオ手に入らないかって問い合わせがくるだろうし，ビデオ配布ってわけにはいきませんか？」
か「えっ配布してくれるのですか？　いいですよ」
Ｋ「いや，そうじゃなくて，そちらに申し込めば手に入るって載せていいですか？」
か「げっ，そんなことしたらいったいどのくらい応募があると思っているのですか？」
K「見当つきませんが，少なくて20，30本……」
か「いやっ，少なくてはどうでもいいんですよ，多くてどのくらいだと」
Ｋ「200，300本……」
か「うそばっかり。今度はX68000ユーザーでなくてもいいし，パソコンを持っていなくても申し込めるのですよ」
Ｋ「ヘタすると2000，3000……」
か「でしょ～。だからやめましょうよ」
Ｋ「しかしですね！　アマチュアCGAの普及を考えるとよい刺激になるのは間違いないし，多くの応募が予想されるってのはそれだけみんな欲しがっているってことだし，CGAコンテストのPRにもなるし，次回のコンテストに応募する目安にもなるし，だいたいDOGAっていうチームは，もとからそういうのをやるために発足したのでは……，DōGAプロジェクトの意義というのは……人生楽ありゃ苦もあるさ……」
こうして，全国ビデオ配布計画は，外部からの強引な圧力によって実行されることになったのでした。詳しい申し込み要項は，今月号の「ペンギン情報コーナー」に掲載しています。トラブルを減らすために申し込み方法を厳しく制限しています。ご注意ください。

とにかく言っておかねばならないことがある

◆最近，X68000の存在に疑問を抱くようになりました。現在流通しているX68000のソフトで実現されるものは，X68000でなければできないものではないと私は思います。まあ「X68000でアーケードゲームがやりたい」とか「このゲームはどうしてもX68000で制作しなければならない」という人であればX68000の存在は絶対なのかもしれませんが……。というわけで，いま私は16ビットコンピュータ（CPUは68000）を製作しています。自作コンピュータでなければできないことをやりたいと考えています。さあ90年代のテーマは「自作コン

▲大山幸典（北海道）

◆アセンブラをやり始めてひと月，CLS.Rというものを作った。内容はCLSのようにいきなり画面を消すのじゃなく，スクロールしながら消すもの。こっちのほうが見て楽しいので作ってしまいました。現在このCLS.RはSRAMに常駐してCLSとしたらこっちを呼ぶようにしています。いやぁーマシン語ってホントに面白いですねぇ。
清水　了（16）　X6800PRO　大阪府
◆ヤッター，大阪へＵターンだっ！　で，でもPEKINが，サンデーネットが，梁山泊が遠くなってしまうっ……。電話代が高うつくなぁ，……はぁ。
鯛　富之（27）　X68000　大阪府
◆電話回線のBBSは人気がありますけど，アマチュア無線にもBBSがあるんですよ。パケット通信というものです。でも，開局するには免許が必要です。皆さんも免許を取って開局しましょう。
平尾　直史（20）　XIturbo　大阪府
◆私のAUTOEXEC.BATが龍になって天に昇っていく夢を見ました。これってやっぱりいい夢なんですか？　藤田　明　（19）　X68000　群馬県
◆MS-DOSって自民党のようなOSですね。ところで最近では，MS-WindowsやTRONのように初心者にもわかりやすいものが出てきたのはいいけれどセンスがよくない。B-TRONなんて改造すると隠れキャラで坂村氏の顔が出てきそうで怖い。きっとROMの片隅に焼きついているぞ。
大村　邦嘉（18）　XIturbo　神奈川県
◆数年前は1990年代になって「Oh! TRON」なんて雑誌が出るかと心配していましたが，いまになってみるとなんてことないですね。
小宮　崇（18）　XIC　埼玉県
◆こわいよー，こわいよー，満開製作所がどんどん「マトモ」な会社になっていくよー。TRONよりもこっちのほうがよっぽど怖いよー。
藤原　利治（23）　X68000ACE-HD，XIturboII　東京都
◆PC-9801のソフトをX68000に移植する場合，いちばんセコイのが768×512モードにして640×400の部分のみを使い，余った部分に意味のない模様を入れてゴマかすというのである。私はこれを「ワク」と呼んでいる。でも「ダンジョンマスター」はワクがあるけどやっててワクワクします。
井上　博嗣（20）　XIturboII　三重県
◆「ダンジョンマスター」は最高ですね。X68000用ののRPGでは買いたいものはなかったけど，これはすぐ欲しくなった。
小阪　友裕（17）　X68000ACE　京都府
◆祝！　イースIII，サーク発売決定！　でもイースIIIのデモを見たらなんとなくできて当たり前の移植なような気がしたけど，ぜいたくでしょうか？
中村　伸夫（23）　X68000EXPERT，XIturboII　北海道
◆荻窪さん，ゲーセンのテトリスは面白いが，パソコン版はクソゲーだというのが気に入りました。
北本　信幸（16）　X68000EXPERT　石川県
◆先日友人の家で初めてX68000の「アフターバーナー」をしました。ゲームには興味がなかったのですが，すごく感動し，こんなに楽しいものだとは思いませんでした。
西口　博史（18）　X68000EXPERT　大阪府
◆このたび，ようやくソーサリアンでエンディングを迎えることができました。といっても追加シ

▲坂本秀司（宮城県）

ピュータはX68000の向こう側」だっ！
岩田　泰徳（20）X1twin/turboZ，MZ-80K/2000　東京都

会社でNECのメインフレームのコンピュータを使っているが，数日前上司より2000年になったときのファイルの日付項目の処理について考えておいてくれ，といわれた。つまり，コンピュータとしては年を4桁で持っているのだが，ファイルのデータ上では年2桁月2桁日2桁の6桁を持っていて，それに基づいて多々の処理を行っており，2000年になると年が00になる，大小関係が逆転してしまうのだ。パソコンにおいても，この大小関係で処理を行っているものが多いと思う。あと10年あるのだが，どうして処理しようかと考えている今日このごろである。
勝岡　義成（27）X68000ACE-HD，MZ-80K/2500　三重県

◆昨年春，突然の落雷で，本体及びディスプレイが入院してしまった。こういうのは保証がきかないそうで，えらい出費になってしまった。あとで，他誌で見たところによると，パソコン保険なるものがあるらしい。入っておけばよかったが，めったに故障などあるわけではなし……。
　まあ，雷が鳴っているときは電源は切っておいたほうがいいですよ。
今井　洋祐（42）X68000ACE-HD　神奈川県

ぬ，ぬお。もう少し，あと1カ月と数日でよいのだ。まだ眠りから覚めないでおくれ。我がゲーマーの血よ。我がプログラマーの魂よ。
西山　新志（18）X1turbo，MZ-80B　福岡県

◆仙台電子専門学校のCMはうそだ。仙台にあんなところはないし，外人が制服を着て自転車に乗っている姿など見たことがない。モルモン教の外人2人組は自転車に乗っているが……。今すぐJAROに電話だ！
坂本　秀司（18）X68000ACE，X1turboII　宮城県

all that's Bug'89

9月号

P.53　スーパーワイドコピー
　リスト2に誤りがありました。131行のror.lをrol.lに，156～170行までの“beq”を“bgt”に変更してください。また，作者名のところで，共同開発者である芦谷知二さんのお名前が落ちていました。お詫びいたします。

P.82　画餅AMA-25h
　上下左右反転で指定領域が破壊されるという症状がありました。
　　DFEF　6E → 6C
　　E015　FD → FA
　　E030　6E → 6C
　　E057　FD → FA
に変更してください。

10月号

P.62　X68000 マシン語プログラミング
　リスト6の268行の，
　　bne wopen0
は，
　　bpl wopen0
の間違いです。訂正してください。

P.74 ショートプロぱーてぃ
　X1用のリストの一部に誤りがありました。20700行のPRINT以下を，
　　PRINT STRING$(39, &H87);
に変更してください。

11月号

P.111　X68000マシン語プログラミング
　リスト5の54行の，
　　bsr getarg
は，
　　bsr nextarg
の間違いです。訂正してください。

P.116,117　マシン語カクテル
　MZ-700のスクロールプログラムで画面桁数が誤っていました。
　　LD BC,79 → LD BC,39
に訂正してください。
　また，P.116の右段18行目の(テキスト)と(アトリビュート)は順番が逆になっていました。正しくは，次のようになります。
　　$D800_H - D000_H = 800_H$
　（アトリビュート）（テキスト）

P.123　MZ-2500グラフィックエディタ作成講座
　印刷ウィンドウでMZ-1P17用の色設定に不備がありました。
　FE06H　20 01 7B 3D 20 02 7A C9 79 C9
　　→　　28 05 3D 20 03 7A C9 4B 79 C9
に変更してください。

P.140　LIVE in'89
　オブ・ラ・ディ，オブ・ラ・ダで前奏部分の音が1音違いました。
　240　p(1)="f8fff8fff8e8e-8d8"
に変更してください。

12月号

P.97　X68000マシン語プログラミング
　サンプルリスト4-bのスタックフレームのためのレジスタ復帰部（13行）が誤っています。適当に直してください。

◆XCについて。
次のプログラムを実行させるとどうなるか。

```
#include <stdio.h.>
  int i=0;
  void main()
 {
  while(i <10) {
        printf(" %d  %d¥n",i,i++);
              }
 }
```

0，0，1，1，2，2，……9，9となりそうだが，1，0，2，1，3，2，……，10，9になってしまう，注意しましょう。コンパイルされたリストを見るとなぜだかわかる。
石原　学（18）X68000　東京都

◆3月24日。ついにファルコムが動いた。T誌によれば，今後についてファルコムは「うーん，そうですね……」とのこと。いまこそX68000ユーザーのPOWERを見せつけるのだ。行け！　目標5万本（これだけ売れりゃあだいじょぶだろ）！
大内　雅雄（17）福島県

◆今回で私，もし載せてもらえていたならば，自画像掲載5周年となります。イラスト常連の方々

ナリオではなく，最初の15本のシナリオです。とにかく面白いゲームでした。ソーサリアンバンザイ！　青島　一高（22）　X1turbo　静岡県

◆私は「銀河英雄伝説」より「宇宙一無責任男」のほうが好きだ。
青山　尊士（18）　MZ-2200　広島県

◆大きな声では言えないが私が東京に就職した理由は「コミケに行きたいから」だ。とても親には言えない……。
百田　浩士（20）　X68000ACE，PC-8801mkII，PB-100　大阪府

◆Zガンダムの森口博子とドラグナーの山瀬まみ。日本サンライズとお笑い元アイドルタレントの間にはなにか見えない赤い糸があるのかもしれない。
二井　吉一（20）　X68000ACE-HD　兵庫県

◆大学の学務係にあった新入生募集のポスターには，大きく「CYBER SPACE」と書いてある。こんなのをあちこちに貼られたらうちの大学はオタク学校だと思われてしまうんじゃないかと心配している。ちなみに僕は九州工業大学情報工学科です。　今戸　肇（20）　X68000　福岡県

◆「もしも学校が……」というドラマ覚えてますか？　もし，パソコンでUFOや宇宙人と交信できたらいいと思いませんか？
中里　晋作（19）　神奈川県

◆ずっと行方がわからなかったフジテレビのMZ-2000ですが，どうやら最近は「あっぱれさんま大先生」に出演しているようですね。
宗片　陽一（20）　X1C，PC-1480　山形県

◆毒物飲料ばやりのせいか「ビタCZ」怖くて飲めないぞ。ところで「ライスサワー88」いかがでしたでしょうか？　今度は「徳山ソーダ」を送りたいと思っています（ウソ）。
藤沢　邦昭（20）MZ-1500/2500，PC-E200 山口県

◆最近長男（5歳）がやたらとゲーセンに連れていけと騒がしい。なにをやるのかと思えば車（ドライブ）のゲームの前に座って真剣になってハンドルを回している。100円入れても30秒もたないのに。挙げ句の果てにわが家のX1turboでそれを作れとせがむ。以前，某テレビ番組のアニメモドキを作った記憶が子供心にもあるようだ。また，忙しくなりそうだ。
藤本　智弘（29）　X1turbo　東京都

◆卒業文集に「これからCを覚えたい，次のXに期待する」と書いてしまった。ちょっとアブない香りが漂ってステキかな？（実はただのバカ）赤いバラはつかなかった。
松本　司（18）　X1G　埼玉県

◆ある日の会話。友人「アルガーナ，あるがーな（2月号P.25）」。私（まずい，反撃せねば……）「アルビオンがあるびょん」。かくして戦いは引き分けに終わった。
河井　啓一（20）　X68000PRO-HD，X1/C/turbo/turboII，MZ-1500　大阪府

◆僕「あのぉ，ボクこんなところにくるの初めてなんですけど……」。相手「あら，でも，ちっとも怖がることはないのよ」。僕「ここに入れればいいんですか？　ちょっと怖いなぁ」。相手「怖がることはないのよ。落ち着いて入れてね」。……2月18日，僕は大人になった（う一む，選挙は緊張する）。
間島　謙一（20）　X68000EXPERT，X1turboZ　東京都

には及びもつきませんが，私なりにがんばったということで……。死ぬまで続けるつもりですのでよろしく。

ともあれ，私もとうとう大学卒業です。これが載っているころには，まがりなりにも教師となっているでしょう。実は私は大阪教育大学にいたのだよ！　三重県の中学生たちよ！　君たちがこれを読んでいるときには私は君たちの先生となっているかもしれないのだ。覚悟しておきたまえ！（でももしかしたら県立上○○○高校かもしれない……）　酒井　強（22）X1　三重県

◆MZ-2500とX68000を持っていながらメインマシンにMZ-2500を使っているのは僕ぐらいじゃないでしょうか？　8ビットあっての16ビット。8ビットをおろそかにしてはいけません。とくに僕の悪友でX68000を持っている人が2人いますが，彼らはMZ-2500を「最低！」などと言ってバカにするので「おまえら，68000って何ビットか言ってみろ！」と言うと「3ビットだ」とか「32ビット」と言ってます。てめーらにはパソコンを触る資格はねぇー。ファミコンで十分だ。

鈴木　武虎（16）X68000EXPERT，MZ-2500　愛知県

◆私が最初にさわったパソコンは，知る人ぞ知る名機，SORD社のM5だった，このパソコンのBASICは超強力で，BASICレベルでの割り込みは簡単にかけられるし，スプライトは命令一発で自由自在に動かせるして，まったく最新の（？）BASICに比べても見劣りしないものだった。ただ，惜しむらくはメモリが妙に小さく（2Kバイト，最大32Kバイトだが，拡張ユニットが本体より高かった），その強烈なハード＆ソフトを生かすことができず，リンボの谷間へと消えていった……。

いま，メモリはM5の1024倍，グラフィックもミュージックも比べものにならないほどすごいX68000を見るにつけ，「ああ……SORDがつぶれず，M68（オフコンであったような気もするが）が出ていればなあ」と思うのである。

畦地　真太郎（19）兵庫県

◆最近のパソコンゲームソフトは高い！　特にX68000のだけ他機種より高いというのは許せない。流通の最大手の日本ソフトバンクさん，なんとかせぇ！

高瀬　昌一郎（22）X1turboZ，X68000PRO　東京都

X1で学習リモコン（未完成ですが）ができることがわかりました。以前X68000でやっていたようなもので，ハードは一部手直しするだけです。Z80の4MHzでスピードが足りるのだろうかと心配されていましたが，なんとかなりました。また，VTR（HSR-5000）のコントロールプログラムも作りましたが，まったく飾りつけはしていません。どうすればよいプログラムといえるようになるのでしょう。

金見　春彦（20）X1turboZ　東京都

◆言いたくはないが，他誌などでの「交換したい」のコーナーを読むと腹が立つ。PC-8801をX68000と交換してくれだとか，X68000を2万円以下で売ってくれだとか。「どこの世界にPC8801とX68000を交換するアホがいるかって！」それをまた取りあげるほうも取りあげるほうだが。

小松　一典（27）神奈川県

大馬鹿者ー！ あーすっきりした。このまえ，某LAOXの店で，FM-TOWNSが私のこの美しい顔に（おい，だれだ？　笑っているのは……）傷をつけやがったのだ。　どーして，CDが回っているときにフタが開けられるような構造なんだ？　おかげでとんできたCDがこの美しい（こだわる）顔にぐわ〜っと！こんちくしょー。

木原　直也（16）X1C　茨城県

◆コンピュータ事業部（奈良）ではいったい何をしていたのだろうか？　MZの冠をあっさりと捨て，AXという大勢のなかへ隠れてしまった。どうして「スーパーMZV2」に続くものを作ろうとしないのか？　MZを日本のパソコン史から消すのか？　私はMZの皮をかぶったAXはいらない。事務機器のようなAXはいらない。私がほしいのは真のMZである。X68000もほしい。だが私が本当にほしいのはコンピュータ事業部という「コンピュータのプロ」が作ったMZという名のパソコンなのである。とにかく，MZなのである。

向井　寛（20）MZ-2500　神奈川県

▲杉本秀昭（宮城県）

▲中城康伸（大阪府）

▲清水健年（東京都）

▲酒井　強（三重県）

▲金子元子（愛知県）

▲安部二郎（東京都）

Oh!Xに関してひと言

◆私は最近流行の「同人ソフト」と呼ばれるものを作っている者のひとりです。といっても私は絵かき専門なのですが……。Oh!Xではそういう記事がまったく見られませんがやはり毛嫌いされているのかしらん。私が思うに，ソフトハウスのサポートの弱いシャープユーザーこそ，お互いの向上のためにも，同人ソフトを通じた情報のやりとりがあるべきだと思うのですが。やはりそのテの雑誌ではシャープ系の影は薄いようです。プロのできない冒険を私たちはこれからもやっていきたいと思ってます。どうか，Oh!X誌上で取り上げていただきたいと思います。もちろん独自の視点から見ていただきたいのです。たまにはそんな特集もよろしいのでは？

吉田　央（20）X68000ACE-HD，X1turbo　奈良県

◆マシン語魔神語　連載　20回
拝啓マシン語によろしく　4回
マシン語体操1-2-3　30回
Z80マシン語ゲーム工房　7回
マシン語カクテル inZ80's Bar　8回

おっと，特集

83年6月「マシン語への招待」
7月「マシン語プログラミング」
85年1月「初めてのアセンブラ」
11月「マシン語“入門”大全集」
87年6月「マシン語プログラム“開発”入門」
89年2月「マシン語“でじたるざんまい”」

などとずいぶんマシン語してきましたが，いまだに私はマシン語でプログラムを作ろうとしていない。今年こそマシン語でプログラミングするぞ。

丹羽　直志（35）X1C/turboZII　愛知県

60歳を越えて パソコンなるものに接している。老化現象の防止にはよい。X68000に関する教科書のような書籍が少なすぎる。その点ではOh!Xは最適なる月間誌といえる。年のせいもあろうが，X-BASICがなかなかわからない。取説などに基本は書いておられるが，不親切な点が多い。手とり足とりの説明がほしい。そこでお願いがあります。Oh!Xに1ページでも2ページでもよい，これ以上やさしく説明はできないというほどの説明と例題をたて説明をしていただきたい。よろしくお願いいたします。関係者のますますの活躍を念願いたします。

佐藤　馨（61）X68000ACE-HD　香川県

Oh!Xイラスト大賞

▶高橋哲史

やってきました。「年に1度だけイラスト投稿者が大威張りで誌面を独占できるぞわあい楽しいなイラスト大賞だー」のコーナーです。あ，初めての人，飛ばして読まないでくださいね！

でもってこの企画も今年で4年目！　うーん，毎年のことながら目頭が熱くなりますね。これもひとえに私の人徳のおかげでしょう！（おいおい）それでは誌面も狭いことだし（ううつらい）さっそくいってみましょう！

第5位（今年も最初は2枚から）

上田　修　大山幸典　加藤信夫　川島祐一
小林貴洋　小松恭郎　笹川明大　佐柳隆行
杉浦　豊　高橋哲史　宮本康司　安川　実

まずトップを飾るのは三重県の上田さん。3月号のカラーにはまいりましたね。そして次は実力派の大山さん。電脳倶楽部でのCGもよかったですね。次は古参の加藤さん。常連の移り変わりの激しい昨今ですがこれからもよろしくお願いしますね。小林さんのイース＋バス◯ードの併せ技は凄かったですね。そしていつも楽しいネタで攻めてくる小松さん。これからもその調子でGOGO！　笹川さんのかわいい絵(というと失礼かな？）もいいですねー。そして超実力派の佐柳さん。うーん，どうすればこんなにうまく描けるんだろう……。そして去年も5位だった杉浦さん。来年度は上位目指して頑張りましょう！　それから実は牛以外も描ける高橋くん。夏には本出しますのでよろしくね（こらこら宣伝すな)。宮本さんの絵って相変わらず渋くていいですね。いまさらですがX1の6MHz化おめでとうございました。安川さん，上達が早いですねー。その調子なら来年度は……。

第4位（3枚載ればもう常連）

伊藤大地　伊藤浩克　岩本智雄　福原　徹
見浦　崇　村山　聡　山崎潤一　山田純二

おおーっともう4位の発表だぁーっ。まず本職はバリバリのプログラマの伊藤さん。イラストもいい味出してますねー。香川県の伊藤さんもどんどん上手くなってとっても楽しみ。お次は新顔の岩本さん。これからもよろしく♪　それからスタッフだかイラスト投稿者なんだかさっぱりわらない川原……じゃなかった福原さん。お世話様ですぅー。そして車のことならの見浦さん。3月号のネタもよかった。そして最近はご無沙汰の村山さん。頑張りましょう！　おっとこれは一昨年度チャンピオンの山崎さん。仕事との両立で大変でしょうネ。そして今年度はペンギンのカット描きでも活躍の山田くん。いまとなっては1987年12月号のあのイラストも懐かしい限りですね(覚えてる？）。うーん，今年は結構スタッフが乱入してるなー。

第3位（4枚載ったらもうハ・マ・リ！）

伊藤健文　大野真実　高橋弘幸
田村憲生　堀　幸司　鶴見明子

おっとこれは実力派の伊藤さん。最近あまりお見かけしませんが復活のご予定は？　それからファンキーなアイデアが冴える大野さん。エセゲームシリーズ（勝手に命名しちゃってすみません）はもうひとつの芸術ですね。次はとっても努力家の高橋さん。コンスタントな投稿姿勢には頭が下がります。ファイト！　おおーっと次は御三家の田村さんだあーっ。内輪ネタですが引っ越しおめでとうございました。次はCG職人の堀さん。だからOh!XにもCG投稿しましょうってば！　それから次は，きゃー鶴見さんだ鶴見さんだーっ。今回のイラスト大賞紅一点！　嬉しいなあ　ほんと一度編集部に遊びに来ませんか？　いいとこですよ，花ゆめもあることだし（笑)。

第2位（5枚！　またその筋に一歩近づいた）

味野真一　小井田伸雄　杉本秀昭　藪田俊平

まずはみゃあさんこと味野さん。端正な絵がとても目を引きますね。そして岩手の元気少年小井田君！　最近は没が続いているようですがこれからも頑張りましょう。お次は実力派の杉本さん。女の子しか描かないのはポリシーですか？　それから藪田さん。いつも丁寧な仕上げで2位と言うのは当然でしょう。さあすると今年の第1位はいったい誰だったのかなぁ⁉

第1位（なんと掟破りの9枚！）

丸藤　俊之

おおーっとこれはすごい！　2位になんと4枚もの差をつけて堂々の1位だぁーっ。ということで圧倒的な実力を見せつけつつ今年度の大賞受賞者は神奈川は横浜の丸藤さんに決定しました。どうもおめでとうございます，パチパチ。丸藤さんのすごいところはその画力もさることながら，やはりゲームに対する思い入れの深さにあると思います。だからあんなに素敵なイラストが描けるんですね。ブロ◯ィアもよかったですよん♪

さて今年もイラスト大賞楽しんでいただけたでしょうか？　そろそろ閉会です。今年の掲載者はなんと88人でした（集計がきつかったあ)。なお5位までのみなさんにはOh!X特製だけど別にたいしたことない記念品が送られる可能性がありますのであまり期待しないで待っていてくださいね。それでは今年も皆さんからの楽しい作品をお願いしますね！　あ，ちなみに来年も当然このページは存在します。ふふふのふ。

（いつの間にかスタッフ入りの高橋哲史）

▶丸藤俊之（神奈川県）

◆Oh!Xは，アマチュアプログラマが活躍できる数少ない雑誌のひとつになったと思います。ピコピコゲームを載せてくれる雑誌は，ここときベーマガぐらいしかありません。これからも，この姿勢を続けてほしいと思います。

ひとつだけ気になることがあります。それは，特集が難しすぎるんじゃないかということです。1985年11月号のOh!MZのマシン語入門は非常にわかりやすかったけど，最近のは，これで初心者がわかるのかなー？　という感じです。〜入門というのは初心者のためにあるのだから，もっとわかりやすくしてほしいなと思います。

最後にゲーム紹介は，ここが最高です！　悪いものを悪いと言うのは大事なことだと思います。

荒田　圭哉（15）X68000ACE-HD，X1C/turboZII，福岡県

◆Oh!Xは，年月が経つにしたがい，難易度が上がってきているようですね。このところついていくのがやっとです。

思えば昔，Oh!MZだったころ，S-OSのできる前はTOOLとHARD全盛の時代でした。単音しか出ないMZへの三重和音プログラムとか，マシン語モニタ改造プログラムとか，FDドライブを自前で安く作ろうとか，いろいろ新鮮な記事がたくさんありました。マシンの奥底のほうをいじくるというプログラムが，あのバリバリの製本のページの上にあふれていました。でもいまは，マシンをいじくるというより，ソフトを買い，それを使って何かを作り出すという記事が増えてきました。パソコンの進化に伴ったOh!Xの変化なのでしょう。時代に乗り遅れかかっている私ですが，今後もなんとかついていこうと思います。でも，ちょっとだけ，昔懐かしいプログラムとか載せていただけたら幸せに思います。

若いみそらでこんなことを考えてしまうのは，やっぱり私が受験生であるからでしょうか。受かったら，X68000を買いますのでどうか今後ともよろしくお願いします。

越智　一秀（18）MZ-2200　広島県

かつてのドラゴンも最近甘い。自分のレベルアップのせいではないし……。もっともっと硬派でもいいと思う。メインを98に替えても，コンピュータのことを学ぶにはOh!Xがいちばん。ほかの雑誌はほとんど市販ソフトの紹介誌になりさがってしまった。これからもしっかりやってください。あの，ソースリストはずっと載せつづけて下さい。いろいろな点で参考になります。“その筋キーホルダー”を持つ一読者より。

山森　一人（21）MZ-2000/2500，PC-9801RX2　石川県

◆ここで一句。

パリパリが　やっとなおった　Oh!X。

清水　達朗（21）X68000ACE，X1，MZ-1200　岐阜県

◆いまこのアンケートを前回のアンケートを送って当たったOh!MZシャープペンシルで書いています。クリップの部分が折れて取れてしまいましたが，まだまだ使わせてもらってます。Oh!MZを初めて買ったのは7年前，MZ-700ユーザーだった頃でした。今まで数多くのパソコン雑誌を書いましたが，現在必ず買っているのはOh!Xだけです。私はOh!Xを2，3年後に役立つ雑誌だと思っています。買

ったときは興味のなかった記事も，2〜3年経ってから読んでみると面白いということもありますし，それに，発売当初ではついていけない記事も2〜3年経てばなんとか理解できるようになることもありますから，これからも，役に立つかどうかはわからないけれど買いつづけていくことでしょう。最後に，今回もぜひOh!Xシャープペンシルが欲しいです。これからもOh!Xの「我が道」を行ってほしいと思います。

高橋　守（19）X1turbo，MZ-1500　千葉県

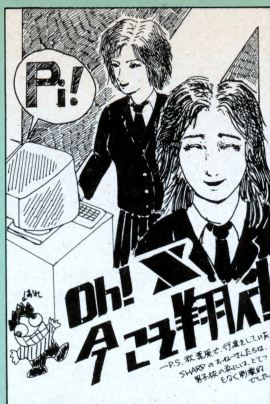
▲若松孝明（栃木県）

▲星野健一（千葉県）

▲見浦　崇（長野県）

近所のパソコンショップレポートだよ〜ん

◆正月に実家に行ってきた。僕の実家は韓国のソウルである。日本に来てX68000を買ってからパソコンに興味を持つようになったことで，やはり自国のパソコン事情が気になりいろいろ調べてみた。韓国にはIBMとそのコンパチばかりで（日本の98ぐらいかな？）すごくがっかりしてしまったが，なんとX68000を売っている店が1軒あったので驚いた。店の話によるとソウルのX68000ユーザーは25人ぐらいいるらしい。でも日本の定価の2倍もするからみんなすごいマニアかも。ほかにもAmiga, Macintoshの店が少しあった。IBM386のコンパチマシンなんか結構安かったがX68000に会ったからそんなもの欲しくもなんともなかったし，IBMを使ってた友達を説得してひとりはAmigaに，またもうひとりはMacintoshに転向させてしまった。さすがにX68000は高すぎたから……。コンパチつくるのならIBM，98みたいなしょうもないものじゃなくてNeXTかMacintoshを作ってほしいな。

話は別だが韓国の電話会社はパソコン通信が盛んになるのに合わせて通話料をものすごく上げてしまったのである。日本でも韓国でも電話会社って儲けることしか頭にないみたいなのが悲しい。

金　永伯（25）X68000　茨城県

函館地区には数軒のパソコンショップがあるが，どの店もやたらとハードばかりで，ソフトなんてないに等しい，彼らは，もっぱらハードの売り上げに熱中し，ソフトなんかくそくらえの方針みたいである。ほんのご愛敬ぐらいに，ほこりをかぶった数年前のソフトがうさんくさそうに置かれている。新しいソフトは，もちろんすべて取り寄せになるのである。

ユーザーの立場からすれば，○月○日発売！　というと，わざわざ2回も足を運ばなくても，ファミコンソフトみたいに1度でその日に買いたいのです。まあ，売り切れならしかたないけど，はじめっからほしけりゃ取ってやる方式の，おこがましくも「パソコンショップ，マイコンセンター」なるものが，山ほどあります。わが函館に。

瀬戸　浩行（31）X68000　北海道

◆私のよく行くショップは新潟県は長岡市，カネキ電です。X68000の売上台数県内トップ！　というこのお店は，こぢんまりとした家電屋というおもむき。ところがところが，ここはパワーユーザーとマネーパワーユーザーのうろつくダンジョンなのです。

98を売っている店の多くは良くも悪くも売りっぱなしのところが多いのですが，ここは違う。客自体が，98にあきたらず集ってきた人が多いのに加え，店主のサポートがすごいのです。定期的にユーザーまわりをし，システム環境のセッティングチェック，自作ソフトのテクニカルレポート，ニュースの提供等々，販売後のサポートがいたれりつくせり。私も気づいたら，X1turboからX68000ACE＋増設1Mバイト＋20MHDユーザーになっていたという商売上手。その上，あるとき払いの催促なしとくれば，どうしますダンナ？

松尾　和浩（28）X68000ACE，X1turboII，MSX　新潟県

特集関係の特殊な話

Oh!Xにはアンケートハガキがついている。アンケートハガキにはいくつかの項目があり，記事作成の参考にされたり，STUDIO Xやリスト下のハミダシに使われたりする。たまに「Oh!Xにハガキを送ったら，○○からダイレクトメールがきた」という苦情がくるが，Oh!MZ時代からのハガキはすべて保管されているので，そういったことはありえない（もっとも，多くは某所にある倉庫の奥に眠っているのだが）。

さて，図1は1989年度のとじ込みアンケートハガキから，毎月それぞれ300枚弱を無作為に抜き出して「今月号の特集について」という欄に書かれている内容から，「わからない」または「難しい」という内容のメッセージを抜き出した場合，どういう結果になるかを示したものだ。ちなみに「300枚弱」というのは枚数を正確に数えたのではなくて，適当に束にしたハガキを560g分ずつ選んで，そのうちのひとつを数えると290枚ちょっとだったからだ。誤差は気にしないようにしよう。

ダントツは11月号で特集は「micro Computer入門」だ。CPUを作るという前代未聞の記事が主な要因だろう。最初は簡単な足し算機を作ろうということで進んでいた企画がエスカレートしてしまった。ところが，締め切り過ぎまでハンダ付けが終わらず，完成して動作チェックと原稿の入稿が同時進行だった。まあ，ちゃんと動いたからよかったが。加えてEDSACもかなり力を貸していると思われる。

第2位は1月号のハードウェア特集だ。ハード関係はどうも難しいというイメージがあるらしい。その一方でコンピュータはハードとソフトで一体なので「もっとハードを」という声も多い。やっぱ，プログラムを開発するときでも回路図が読める人は強い（よほどのことがないと必要はないが）。

3位は7月号の3Dグラフィック。Zバッファなどでは数式を抑えて図を多くしたにもかかわらず，題材自体に馴染みがないためか，上位にくい込んだ。丹氏独自のプログラムですべて整数化して計算するため，最近の流行とは相容れない部分もあったらしい。「正規化されていないのはなぜか」という鋭い指摘もあった（前後関係が不正確になる）。

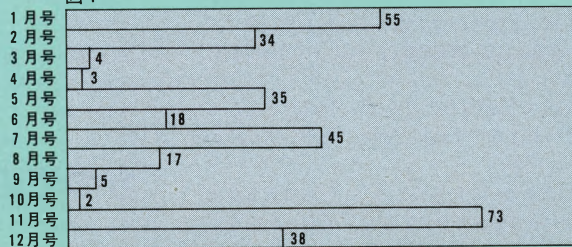
これら難解度上位の特集では逆に多くの「わかりやすい」という声も同時に聞かれるのが特徴だ。もちろん「基本的な部分だけだからつまらない」とか「突っ込みが足りない」という声もある。

いずれにせよ，これらの「わからない」という声がわかろうとしている過程で発されていることがうれしい。少々わからなくても興味を持って読んでくれる人は多いのだ（甘えてはいけないな）。作ってる側がいってはいけないことだが，多くの読者がOh!Xの特集記事を読みこなしている。これはもしかしたらもの凄いことなのかもしれない。

さて，「わからない」が少ないときの特集で，なおかつ「わからない」がある。これはほとんど「特集の意図がわからない」とか「持ってないんでわからない」というヤツだ。うーん，特に4月号のゲーム特集はブッ飛んでたからなあ……。とりあえずパソコン・マガジンに移籍したN氏のせいにしておこう。

（特集の重く暗い部分担当のU）

●1989年度特集一覧

1月号　いきなり初春からハードウェア
2月号　マシン語“でじたるざんまい”
3月号　BASIC“おもちゃ箱”
4月号　ゲーマーたちの“新深夜族”宣言
5月号　MIDIサウンドデータ料理術
6月号　これからのX family
7月号　3Dグラフィックへの飛翔
8月号　X1プログラミングガイドブック
　　　　3Dグラフィックの深淵へ
9月号　活用ハードディスク＆プリンタ
10月号　ゲーム面白心理学
11月号　micro Computer入門
12月号　Cプログラミングへの招待

ぼくらの掲示板

●掲載ご希望の方は，官製ハガキに項目(売る・買う・氏名・年齢・連絡方法……)を明記してお申し込みください。
●ソフトの売買，交換については，いっさい掲載できません。
●取り引きについては当編集室では責任を負いかねます。
●応募者多数の場合，掲載できない場合もあります。

仲間

★筑紫高宏＆古旗一浩からのお知らせだ〜
X1/turbo, MZ-700/1500, MSX……のユーザー会「EXTRA」ではいままでOh!MZ, Oh!Xに掲載された古旗一浩のゲーム＆ツール＆システムを配布します。いままでダンプリストを避けていたユーザーもEXTRAに入会すればOK！ ちなみに発表できなかったソフトも配布しています。特にSystem-7Bの掲載されたOh!Xは品切れですので入手したい人はぜひご入会を。主な活動の内容は会報の発行で，会員の数は現在50人程度です。さらに今回Picture Compilerというコンパイラを開発しました。これでなにができるかというとOh!X 2月号掲載のEyelarthのようなものが作れます。というわけでこのコンパイラも配布しますのでMZ-700/1500ユーザーはぜひご参加を（MZ-700/1500ユーザー以外でも興味のある方や移植したい方は大歓迎)。コンパイラはマニュアルとソフトで1,500円です（マニュアル1,000円，ソフト500円)。コンパイラは入会しなくても配布しますが，そのほかのゲーム＆ツールは入会しないと配布しないのでよろしく。〒811-42 福岡県遠賀郡岡垣町戸切794-3 筑紫高宏
★サークル「白竜亭」ではX68000ユーザーの会員を大募集中です。活動は月に一度の会報の発行と情報交換など。あと皆さんが参加できるような企画も考えています。興味がある方は62円切手同封のうえご連絡を。〒946 新潟県北魚沼郡小出町古新田447 上村一宏（17)
★X68000ユーザー対象のサークルを愛知の三河地域で作りたいと考えています。内容は情報やPDSや自作ソフトの交換など。スタッフおよび会員を募集しますのでX68000を持っているヒマな方はハガキまたは封書でご連絡を。〒442 愛知県豊川市下長山町岩下96-1 戸田史朗（25)
★私たちはディスクでの会報（月1回）によるプログラムの交換および配布を中心に活動を行っています。現在会員は約50名です。プログラム作成や研究が好きな方のご参加をお待ちしています。もちろん初心者の方も大歓迎します。対象機種はX68000です。興味のある方は250円切手を同封のうえご連絡を。折り返しディスク会報の創刊号と案内書を無料でお送りします。〒260 千葉県千葉市真砂3-10-11 鈴木淳（17)
★このたびKORGのM1とX68000をフルに活用したオリジナルアルバム「prophecy vision」を作りました。46分テープで10曲入りです。このテープを送料＋500円でお分けしますので，希望者は500円切手と住所を書いた返信用封筒を同封したうえで封書で送ってください。〒260 千葉県千葉市都賀の台2-4-4 稲家克郎（17)
★X68000のPDSを探しています。X68000ユーザーで「よし，PDSの交換をしてやろう！」という方の連絡を待っています。〒671-12 兵庫県姫路市勝原区山戸241-10 山根邦博（16)
★S-OSクラブ「Illegal」では会員を募集しています。S-OSでのプログラムの開発，情報交換，月1回の会報発行が主な活動です。対象はS-OSの発表されている機種のユーザーおよびそのほかの機種でS-OSを開発したいユーザーです。連絡は往復ハガキで。使用機種，年齢，電話番号を明記のこと。〒064 北海道札幌市中央区南20条西7-2-20 渡辺裕之（17)
★フォント研究機関「タイプ・ラボ」では統合化フォントシステムを構築するにあたり，開発スタッフを募集します。また，いままでに開発したフォントを収録した全集も用意しています。連絡は62円切手同封のうえ封書で。〒910 福井県福井市渡町358-4 平木敬太郎(22)

売ります

★48ドット熱転写カラー漢字プリンタ「CZ-8PC4」（新同キズなし，箱・付属品あり）を送料込み4万5千円ぐらいで。連絡は往復ハガキで。〒221 神奈川県横浜市神奈川区広台太田町2-4-124 須川英樹（19)
★24ドット熱転写カラー漢字プリンタ「CZ-8PC3」（黒色，新同，箱・付属品あり)を3万5千〜4万円で。連絡はハガキで。〒661 兵庫県尼崎市南武庫之荘10-60-6-202 宮崎直樹（21)
★エプソン「VP-2050」(「CZ-8PG2」同等品，信号ケーブルなし，白色）＋カットシートフィーダ「VP-2000CSFW」を10万円くらいで。値下げ可。連絡は往復ハガキで。〒241 神奈川県横浜市旭区白根8-22-8乾荘2-B室 池田健一（23)
★X68000用カラーイメージユニット「CZ-6VT1」を2万円，数値演算プロセッサボード「CZ-6BP1」を3万円，FAXボード「CZ-6BC1」を3万円，カラープリンタ「CZ-8PC3」を2万円，モデムユニット「CZ-8TM2」を2万円で。連絡は往復ハガキか62円切手同封のうえ封書で。〒025 岩手県花巻市鍛冶町4-15 清水啓嗣（28)
★カラーイメージボード「CZ-8BV1」を4千円程度で。FM音源ボード「CZ-8BS1」を7千円程度で。どちらも箱・マニュアルなど付属品はすべてあり。連絡は希望価格明記のうえハガキで。〒336 埼玉県浦和市岸町1-2-9 山口明徳(19)
★エプソン製ハンディカラーイメージスキャナ「GT-1000」（完動，美品）を送料込み4万5千円で。連絡は往復ハガキで。〒899-71 鹿児島県曽於郡志布布町安楽215-3 南正治（30)
★X1turbo用漢字第2水準ROM「CZ-8Bk3」を6千円で。連絡は往復ハガキで。〒171 東京都豊島区要町1-46-10北村方 木村哲也（21)
★X1用マウス「CZ-8NM2」（本体のみ）を2千円以上で。連絡は往復ハガキで。〒617 京都府長岡京市天神3-11-12 三戸詳司(19)

買います

★X68000用MIDIボード「CZ-6BM1」を1万3千円以内で。ローランド「MT-32」または「CM-32L」を2万9千円以内で。完動，付属品付きなら多少の汚れ・キズは可。連絡は往復ハガキで。〒170 東京都豊島区東池袋5-48-15 鈴川正洋(15)
★1Mバイト増設RAM「CZ-6BE1」を1万5千円程度。MIDIボード「CZ-6BM1」または「SM-68M」を1万4千円程度で。ローランド「MT-32」,「CM-32L」,「CM-32P」,「CM-64」をそれぞれ2万5千円，3万円，2万円，5万円程度で。完動，付属品・説明書付きなら多少の汚れ・キズは可。送料当方負担。連絡は希望価格を明記のうえ往復ハガキで。〒357 埼玉県飯能市岩沢184-2 岡田隆裕(18)
★X68000用拡張I/Oボックス「CZ-6EB1-BK」を4万円以内で。アンプ内蔵スピーカーシステム「AN-S100」を2万円以内で。完動，付属品付きならキズあり，箱なしも可。状態と希望価格を明記のうえ往復ハガキで。〒260 千葉県千葉市磯辺3-12-10 山川秀幸（21)
★X68000用CRTフィルター「BF-68PRO」を5千円以内で。ローランド「MT-32」,「CM-32L」を2万8千円以内で。完動，付属品付きなら多少の汚れ・キズは可。連絡は往復ハガキで。〒260 千葉県千葉市都賀の台2-4-4 稲家克郎(17)
★RGBシステムチューナー「CZ-6TUを1万7千円(グレイは2万円)で。完動品ならキズあり,箱・説明書なしでも可。送料当方負担。連絡はハガキで。〒328 栃木県栃木市片柳町2-53-7 山野上敬裕（16)
★X1用RAMボード「MB-1000」(デジック製)を2万〜3万円程度で。完動なら可。連絡は往復ハガキで。〒503 岐阜県大垣市林町7-783-2 宇野靖(21)
★X1用FM音源ボード「CZ-8BS1」＋付属品を8千〜9千円（送料込み）で。連絡は往復ハガキで。〒989-24 宮城県岩沼市相の原2-9-2 加藤充浩(14)
★X1データレコーダ「CZ-8RL1」，拡張I/Oポート「CZ-8EP」，漢字ROM「CZ-8BK2」を各5千円前後で。連絡は往復ハガキで。〒636-03 奈良県磯城郡川西町唐院213 吉仲正和(19)
★X1用カラーイメージボード「CZ-8BV2」を送料込み1万円程度で。付属品付き，完動品のこと。連絡は往復ハガキで。〒652 兵庫県神戸市兵庫区塚本通3-1-4 岡崎一義(29)

バックナンバー

★Oh!X1986年9月号，1989年2月号を送料込み各1500円程度で。切り抜き不可。連絡は往復ハガキで。〒545 大阪府大阪市阿倍野区三明町1-6-8ヤングパレス阿倍野橋303 江角浩行（18)
★Oh!X1988年5月号，1989年2月号を2冊で2千円（送料込み）で。切り抜き不可。連絡は往復ハガキで。なるべく近県の方お願いします。〒850 長崎県長崎市水の浦町29 森貴弘（17)
★Oh!X1988年8月号，1989年3月号を送料込み各千円で。切り抜き不可。連絡は往復ハガキで。〒969-11 福島県安達郡本宮町字馬場104 馬場保幸（15)

どんな悩みもスッキリ解消

ざ・質問箱SPECIAL

解答者　影山裕昭

「言わせてくれなくちゃだワ」と一緒に年に1回やってくるのが，いつもの質問箱の枠を拡大してお送りする「ざ・質問箱SPECIAL」。甘口な質問から激辛の質問まで担当の影山が責任をもってお答えいたします。では最初の方どーぞ。

Q **BASICのファイルネームを変えようと思い，NAME命令を使ったところ，FILES命令でファイルをとったときにはファイルネームが変わっているのに，IPLがロードするときには，ファイルネームが元に戻ってしまいました。強制的にファイルネームを戻されてしまったのかと思い，もう一度ファイルをとってみるとファイルネームは変えたときのファイルネームと同じでした。BASICのファイルネームは変えることはできないのでしょうか。またFILES命令でファイルネームが出ない隠しファイルは作れないのでしょうか？**

千葉県　市原　貴広

A 実に素朴な質問で結構なんですけど，使用機種くらいは明記しておいてくださいよ。ま，おそらくX1だろうということで話を進めていきますからね。

さて解答ですけれどX1のディスクのレコード構成を知っていればなんてことないんですね。以前にもちょっと触れたことがあったかと思いますけどX1のディスクは1280レコードあって，レコード0がIPLに関する情報，レコード1～13がディスクBASIC本体，レコード14がFAT，レコード16～31がディレクトリ領域，レコード32以降がプログラムやデータが格納されるデータ領域となっています。

2HDになってもレコード32以降のデータ領域が増えるだけでほかは同じです。このうちファイルネームやらのファイルに関するもろもろの情報はディレクトリ領域に記録されています。またIPLが読み込むファイルの情報はディレクトリ領域とは別にレコード0にも記録されていて，当然ここにもファイルネームは記録されているのです。先ほどのNAME命令はディレクトリ領域に記録されているファイルネームを変更する命令ですから，FILESではファイルネームが変わっていても，IPLで表示されるファイルネームは元のままなのです。

だからNAME命令を使わなくとも直接レコード0に記録されているファイルネームさえ変更することができれば，IPLで表示されるファイルネームを変えることができるんです。レコード0のファイルネームを書き換えるための命令はBASICにはありませんが，要は直接レコード0を読み込んでからファイルネームだけを変えて，レコード0に書き戻せばいいのです。その方法をこれから紹介しますが，誤った操作を行うとディスクを破壊する恐れがありますので，この変更は必ずバックアップをとったディスクに対して行ってくださいね。

まず，ドライブ0にファイルネームを変更したいディスクを挿入して，

```
DEVI$ "0:", 0, A$, B$
```

としたら，

```
C$="ファイルネーム"
C$=C$+STRING$(13-LEN(C$), CHR$(32))
```

とします。ファイルネームは必ず拡張子をつけずに13文字以内にしてください。そうでないとIPLを破壊します。そうしたら，

```
MID$ (A$, 2, 13)=C$
DEVO$ "0:", 0, A$, B$
```

とします。これで変更は終了しました。ではIPLから起動してみてください。ちゃんと変わっているでしょ。

ところでシークレットファイルを作るには，

```
SET "ファイルネーム", "S"
```

です。これはちゃんとマニュアルに載っているんですよ，もう。

Q **ちょっと質問。X68000のX-BASICで「Ok.」と出るときと「Ok」と出るときがありますけど，これはどーしてですか。**

東京都　尊谷　宏

A こんな細かいところによく気がつきましたね。もしかしたら君はA型なんじゃないか？　ま，それはいいとしてOkのあとのピリオドにはちゃんと意味があるんです。ピリオドがあるときはプログラムの実行継続が可能なときで，ないときは不可能だってことを教えてくれているんです。平たくいえばCONTできるか，できないかってこと。これはX1のBASICでも同じですよ。

Q **これからマシン語を勉強するうえで，どうしても知っておきたいことがあるのですが，プログラムが起動した直後のA0～A4の数値はいったいなんの情報なのでしょう？**

静岡県　堀井　将弘

A A0はメモリ管理ポインタのアドレス，A1はプログラムの終わり+1，A2はコマンドラインのアドレス，A3は環境変数領域のアドレス，A4は実行アドレスを示しています。ひととおりざっと説明していきましょう。A0の指すメモリ管理ポインタとはOSがメモリの使用状況を把握するために必要な情報が置かれているところです。これはプロセスを管理する上で重要な部分で，本誌1990年1月号のX68000マシン語プログラミングで詳しく説明されているのでそちらを参照してください。A2はプログラムに与えられたパラメータの情報が置かれているアドレスを指していて，

```
TEST /S /D
```

のようにOSからプログラムを起動した場合はA2の指す先のアドレスから，

```
05 2F 53 20 2F 44 00
```

となっているはずです。最初の1バイトはパラメータの総数で，2バイト目から00までが与えられたパラメータを表しています。これについても1989年11月号のX68000マシン語プログラミング入門で扱われているので参照してください。A3の指す先はパスなどの設定情報が置かれている領域です。残ったA1とA4については改めて触れるまでもないでしょう。

Q **ついに登場という感じの「超多機能アセンブラOHM-Z80」ですね。私はS-OS"SWORD"を持っていないので直接OHM-Z80の恩恵にあずかるわけにはいかないのですが，やっぱりこういうのが載ると嬉しくなってしまいます。それにしても不思議なのがOHM-Z80自体がOHM-Z80の文法で書かれているようである，という点です。いったいどうなっているのですか？**

広島県　三原　啓志

A 自分でエディタを作る場合を考えてみたらどうでしょう。最初は既存のエディタを使って単純にただ文字を入力できるだけのエディタを作りますね。これをバージョン0.1としよう。で，使っているうちにやっぱりページスクロールくらいほしくなってきます。そうなったらバージョン0.1のエディタを使って自分自身のソースリストをエディットするわけです。このようにいちばん元になる部分だけ作っておいて，あとでだんだんと機能を付加していくような作り方はよくあることです。

実はOHM-Z80以前にも大貫さんは「構造化ASM」という高級言語指向のアセンブラを自作して使用されていました。SLANGなどはそれで書かれていましたね。このソースリストも，アセンブラとしての最低限の機能を兼ね備えている基本バージョンを作っておいて，そのあとはそれを使って機能を拡張していったものでしょう。これだけのプログラムをひと息に作ったりできるものではありません。デバッグを兼ねて，使いながら改良/拡張するというのは効率のよい開発方法です。

Q **X68000の付属のワープロで全角と半角の文字を交互に入力すると「変換中」となったまま止まってしまうことがあるのですが，私のX68000が変なのでしょうか。あとX-BASICから「！」でCOMMAND.Xに行って少しなにかやってEXITでBASICに戻り，さらにEXIT( )，SYSTEMでCOMMAND.Xに戻ろうとするとここで止まってしまいます。2つの現象について原因なんぞお教えください。使用システムはX68000 EXPERT，Human Ver. 2.00です。**

宮城県　坂井　一弘

A この文面からだけではあなたのX68000が故障しているとは考えられません。それより，あなたのシステムが気づかないうちに破壊されていることも考えられるので，一度マスターディスクで立ち上げてシステムとワープロディスクをバックアップし直すことをおすすめします。あとX-BASICから子プロセスとしてCOMMAND.Xを立ち上げたときに，アドレスエラーやバスエラーを発生するようなプログラムを走らせたりすると（あまりないと思いますが），BASICに戻ったときにSYSTEMやEXIT( )でOSに戻ろうとしたときに，「COMMAND.Xの起動に失敗しました」といったメッセージが表示されることがあります。

この2つのどちらにもあてはまらない場合はHumanのバグなのかもしれません。現行のHumanの最新バージョンは2.01です。初期出荷のPROやEXPERTの一部にはVer2.00のHumanが同梱されていましたが，いくつかのバグがあったようで現在同梱されているものはすべてVer2.01となっています。ユーザー登録された人にはバージョンアップ版のシステムディスクが送付されたはずなのですが，まだユーザー登録していないようでしたら早めに登録はがきを出したほうがいいでしょう。いずれにせよ，最寄りのシャープサービスセンターに電話で連絡してみてください。おそらく無償でシステムの交換に応じてもらえると思います。

Q **付属のワープロを使っていて思うんですけど，入力する際にいちいち枠が出てくるのが邪魔なんです。なんとかしてこの枠を出さないようにするよい方法はないものでしょうか。**

東京都　児玉　勇介

A よくあるワープロ関係の質問ですけど，これには常套手段があります。手順としては，まず最初に入力モードを無変換にします。それから全角とひらがなとローマ字またはかなキーのランプをつけます。こうす

ると無変換モードでも一括変換と同じ方法でひらがなを漢字に変換することができるようになるのです。ただし変換キーを押したときには枠が出てしまいますけどね。意外にこれは知られていないらしいけど，ワープロのマニュアルにはちゃんと書かれているんですねー。

Q **私はSWITCH.XでBOOTのROMアドレスを変えてしまった愚か者です。X68000（CZ-600C）が正常に立ち上がらず（「正しいディスクをセットしてください」などのメッセージが出ない）困っています。直し方を教えてください。お願いします。**

埼玉県　町田　友行

A 取扱説明書にも書いてありますが，起動するときにOPT.1キーを押しながら立ち上げてください。すでにおまじないのように使っている人も多いでしょうが，OPT.1キーを押しながら立ち上げるとX68000はS-RAMの内容を無視して，無条件に内蔵ドライブからシステムを立ち上げようとします。ですからなんらかの理由でシステムが正常に立ち上がらなくなったら，あわてずOPT.1キーを押しながら立ち上げるようにしてください。

Q **HDをフォーマットしていくつかの領域に分けて（システムは転送する）から，領域を選択しないで，スイッチを切り，次にスイッチを入れたときに出てくる「領域を選ぶメッセージ」で選ぶと，以後，そればかり選択されてしまいます。そのときに何度も選べるようにできないものでしょうか。使用機種はX68000ACE-HDです。**

北海道　原田　伸宏

A X68000ACE-HDというと付属のシステムはHumanVer1.0Xでしょうか。それでしたらFORMATコマンドで，

　FORMAT/H

としてハードディスクを初期化すると起動時に領域選択のためのメニューが現れるようになります。しかし，HumanVer1.0Xでは起動するときに選択した領域以外をシステムで使用することができないので，40Mバイトハードディスクでも10Mバイトの領域から起動した場合には残りの30Mバイトのデータを利用することができません。できるならシステムはHumanVer2.00にしておきたいところです。

Ver2.00以降のFORMATコマンドでハードディスクを初期化した場合には領域選択で自動起動を設定しなくても，一度電源を切ったあとに最初に選択した領域が自動起動に設定されるので次回からは選択のメニューが出なくなってしまいます。しかし，逃げ道は用意されていて，HELPキーを押しながら立ち上げると，どの領域から起動するか選択できるようになります。

Q **いつもはX1turboを高解像度にして使っているのでキャラクタフォントが綺麗でいいのですが，たまにCZ-8FB01なんかを使うとギザギザした文字になるので，なんとかして綺麗なフォントで使いたい！　X1turboであればCZ-8FB01を使うときも高解像度で使えると思うのですが，はたしてできるものなのでしょうか。もしできるならその方法を教えてください。**

鳥取県　大久保雄一郎

A CRTCと画面管理I/Oを高解像度用に設定すればできます。CRTCのレジスタへはWIDTHで表示桁数を指定するたびに設定されるので，その設定されるべきデータがRAM上のIOCS領域のDDH番地から置かれることになっています。CRTCに設定するデータは日本ソフトバンクから出ている『X1システム研究室』などに公開されていますから，それらを参考にしてできあがったプログラムがリスト1です。これ単体でももちろん実行できますが，サブルーチンのかたちにしてStart up. Basの中に加えておくのが望ましいでしょう。

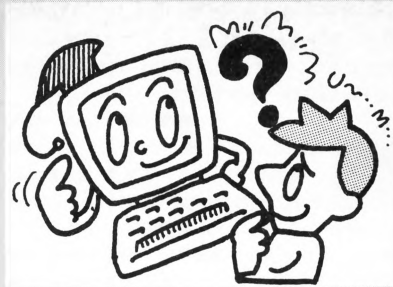
プログラムについて少し話すと，20行ではCRTC関係のI/Oを見ているんじゃなくて，フロントにあるレゾリューションスイッチの状態を調べています。ここではボタンがスタンダードになっていたら，画面管理I/Oを低解像度に設定しています。40行以降がCRTCを高解像度用に設定する部分です。このままですとこのプログラムを実行したときの画面モードが40桁以外の場合は画面が乱れますので，必ず40桁モードで実行するようにしてください。80桁モードで実行したいんであれば40行のA$＝"に続く8文字を「6B505988」に変更してください。

Q **僕はデバッガが逆アセンブルした結果などをプリンタに打ち出して保存しているのですが，プリンタの調子が悪いのかX68000の調子が悪いのか，最近プリンタがきちんと動いてくれないので困っています。ワープロではいままでどおり打てるのですが。一度修理に出したほうがいいのでしょうか。使用システムはX68000ACEです。**

熊本県　稲垣　伸一

A ワープロでプリンタを使う場合は正常に動いているみたいですから，X68000やプリンタが故障しているとはあまり思えません。

ところで，使っているデバッガはDB.Xだろうと思いますが，そうなるとプリンタへ出力するスイッチはありませんけど，いったいどうやっているんでしょう？　たぶんキーボードコントロールを使っていると思うんですけど……。CTRL＋Pを使っているか，もしくはOSのリダイレクト機能を使っているんだと思います。

考えられることとして，最近になって使用システムを変えたことはありませんか。もし以前に使用していたシステムがHuman1.0Xでいま使っているシステムがHuman2.0Xだったりすると，この症状の原因はシステムの変更にあるかもしれません。Human2.0Xは標準状態でデバイスドライバとしてHISTORY.Xを登録していますが，このデバイスを登録するとCTRL-PとCTRL-Nによるキーボードコントロールが無効となるのです。ですから対策としてはCONFIG.SYS中でHISTORY.Xをデバイスとして登録している行を削除するか，先頭に＊をつけて無効扱いする，もしくはコマンドモードから，

　HISTORY /K

としてヒストリの使用を一時中止してください。ヒストリを再度使うときは，

　HISTORY /U

です。

そうじゃなくてHumanのリダイレクト機能を使ってプリントアウトしているようでしたら，設定しているデバイスドライバがプリンタの機種にあっていないかもしれません。どのデバイスドライバがどの機種に対応しているかはマニュアルに書かれているので，詳しくはそちらを参照してください。それでもおかしかったらハードのほうに原因があるのかもしれません。早めに最寄りのシャープサービスセンターに連絡したほうがいいでしょう。

Q **X-BASICでゲームを作っているのですが，エンディングで制作者の名前などを表示させるのはスクロール命令がないので，スペースキーを押したら続きを表示さ**

リスト1

```
10 ' check front switch
20 IF (INP(&H1FF0) AND 1)=1 THEN OUT &H1FD0,0:GOTO 120
30 ' width 40 CRTC data
40 A$="35282D841B00191A000F0000000000000000"
50 FOR I=0 TO 17
60   OUT &H1800,I                           ' CRTC register
70   OUT &H1801,VAL("&H"+MID$(A$,I*2+1,2)) ' CRTC data
80 NEXT
90 A$=HEXCHR$("35282D841B00191A000F0000000000006B505988")
100 MEM$(&HDD,LEN(A$))=A$
110 OUT &H1FD0,67 ' gamen kanri I/O
120 END
```

PRNDRV.SYSのひみつ

私の持っているプリンタはNECのPC-PR406というやつなんですが，これがもう5年くらい前の熱転写プリンタだから，最近ではリボンの入手が難しくなってきました。おかげでもっぱら感熱紙のお世話になっています。

いきなりわけのわかんないことを書いてしまいましたが，X68000でプリンタを使うときに必ず必要なのがプリンタドライバの登録ですよね。そのプリンタドライバ，マニュアルには指定できるオプションがひとつしかないように書かれているけど，実は4つ指定できるんです。ここではそれを紹介しましょう。

一般的に知られているのはMですが，ほかにもW，L，Bが指定できるんです。注意したいのはオプションは複数指定が可能ですが，その際第1オプションは#/の後ろに書かれますが，第2オプションからは/で区切ります。#はいりません（エラーになる）。

では，ひとつずつ順に説明していきましょう。

W：1行に何文字印字するか指定します。指定できる範囲は1～100（半角文字）で，80桁であれば「W80」のようになります。

L：1ページに何行印字するか指定します。指定できる範囲は1～100で，40行なら「L40」でOKです。

B：プリンタバッファに何Kバイト確保するか指定します。このオプションを指定すると並行印字ができるようになります。指定できる範囲は1～100で10Kバイト確保するなら「B10」です。

M：いまさら触れるまでもないでしょう。マニュアルを参照してください。

この例をデバイスとして登録するなら，

　DEVICE＝PRNDRV.SYS #/W80 /L40 /B2

となります。

ちなみにプリンタに出力するデバイスはPRNとLPTがありますが，どちらを使っても同じだと思っている人もいるかもしれません。実際には区別されていてLPTを出力先に指定した場合は漢字IN/OUTコードが送られません。よって漢字を出力するときはPRNを使います。

せる，という方法をとっています。もっと上手に表示させられないでしょうか？（グラフィック画面でもいいです）。使用機種は X68000EXPERTとHuman68k Ver2.01です。

神奈川県　山口　隆夫

A 山口さんはPC-9801やPC-8801にスクロール命令が「ある」とも書いていきましたが，それらはすべてグラフィック画面に対してのスクロール命令です。それなら X68000にもちゃんとHOMEという命令があるではないですか。グラフィック画面でもいいんであれば，これで十分通用します。というわけで，早速ですが HOME 命令を使ったサンプルを紹介します（リスト2）。まずは入力して実行してみてください。画面の下から上へ向かってスクロールしていくのがわかるでしょう。

一応，なにをやっているかざっと説明しておきましょう。90行までは画面サイズを256×256(実画面は512×512)にしてクリッピングエリアを（0,0）－(511,511）に設定します。100 行から先がスクロールのメインルーチンです。ループの中で i を 2 倍しているのはX-BASICの FOR 文にSTEPが使えないからで，このようにして STEP 2 と同じ結果を出すように調整しています。次の 110 行がグラフィック画面の表示位置をずらしてスクロールさせているところです。 i は 2 つずつ増加しますから 2 ドットスクロールになるのはわかりますよね。そのあとでmod 1024をしている理由は自分で考えてみてください。120 行の条件判断がこのプログラムの最重要部分なんですが，この部分も独力で理解してみてください。やっていることは i × 2 が64で割れればSYMBOL 文を実行，割れなければ空ループを回して時間を稼ぎます。注目してほしいのはSYMBOL 文で指定している座標です。実画面サイズと表示画面サイズを頭に入れて考えれば，すぐに理解できると思います。どうしてもわからなかったら40行を削除して，50，60行の / ＊ を取って実行してみてください。最初にやったときに画面に表示されていた部分が横線より上で，それより下が画面に表示されない部分です。140行のループの値を変えてみるのもいいでしょう。

と，まあ，これでも十分なのですが，もうちょっとカッコよくすることを考えてみましょう。X68000はグラフィック画面とは別にスプライトやBGを表示することができるのは知ってますよね。BGは 1 面に最大4096個（64×64）のパターンを定義することができて，画面サイズが 256×256のときは 2 面表示することができます。ここでも画面サイズを256×256にしてBGを 2 面使ったプログラムを紹介しましょう。

BGに定義するパターンは画面サイズが 256×256のときは 8 × 8 と決まっています。しかし，自分でパターンを作っていくのは大変な作業なので，あるものを利用しようということで，スペースハリアーのフォントパターンを吸い出すプログラムを作りました（リスト3）。いわゆるSEGAフォントですね。スペースハリアーがなければアフターバーナーでも結構です。

まず，スペースハリアーを立ち上げてタイトル画面が表示されたらリセットします。その後にBASICを立ち上げて，リスト3を実行してください。実行するとドライブBにFONT.DATというファイルが作られます。このファイルを読み込んで，

　　5 screen 0, 3, 1, 1
　　7 sp_init( )

の 2 行を加えてください。さらに10行を，

　　10 char sp(63) = {

に変えてください。ここまでやったらセーブして実行してください。これでフォントパターンが定義されます。なお定義されるフォントは，スペース，数字，アルファベットの大文字，それと！，．？－＠です。

次にパレットブロックを設定しますのでリスト 4 を入力，実行してください。パレットブロックについては説明しませんので，マニュアルを参照してくださいね。これでBGを使うための準備は完了です。なかなか大変でしょう。

さて，リスト 5 が本命のBGを使ったスクロールプログラムです。言葉で説明してもわからないと思うので，とにかく実行してみてください。ひとつだけ説明しておくと，関数 bg_print の引数は 1 番目からBGのページ番号，X 座標，Y 座標，文字列の順番で並んでいます。ただし X，Y については座標チェックをしていないので，X，Y とも 0 ～63の範囲を超えないように注意してください。それから文字列が長くて表示途中に X 座標が63を超えるとエラーが出て止まります。 2， 3 行の追加でエラーチェックもできるはずですから，興味を持った方はぜひ改造してみてください。

リスト2

```
 10 /*
 20 /* グラフィック画面スクロールサンプル
 30 /*
 40 screen 0,0,1,1
 50 /*screen 1,0,1,1
 60 /*locate 0,15:print string$(32,"_")
 70 console ,,0
 80 window(0,0,1023,1023)
 90 wipe()
100 for i=0 to 2048
110   home(0,0,i*2 mod 1024)
120   if (i*2 mod 64)=0 then {
130      symbol(72,((256+i*2) mod 1024)," X 68000",2,2,1,rnd()*15+1,0) } else {
140      for j=1 to 220:next }  /* wait
150 next
```

リスト4

```
10 dim int c(16)={
20   0,45386,49812,54238,58664,63090,65468,0,
30   0,0,0,0,0,0,0,0 }
40 for i=0 to 15
50   sp_color(i,c(i),1)
60 next
```

リスト5

```
 10 /*
 20 /* BGスクロール サンプル
 30 /*
 40 screen 0,3,1,1
 50 bg_fill(0,256+32)
 60 bg_fill(1,256+32)
 70 bg_scroll(0,0,0)
 80 bg_scroll(1,0,0)
 90 bg_set(0,0,1)
100 bg_set(1,1,1)
110 sp_disp(1)
120 for i=2 to 5
130   bg_print(0, 3+32,i*4  ,"PERSONAL WORKSTATION X68000")
140   bg_print(1, 3+32,i*4+2,"PERSONAL WORKSTATION X68000")
150 next
160 for i=0 to 256
170 /* 横スクロール
180 /*bg_scroll(0,i,0)
190 /*bg_scroll(1,(512-i) mod 512,0)
200 /* 縦スクロール
210   bg_scroll(0,256,256-i)
220   bg_scroll(1,256,(255+i) mod 511)
230 next
240 end
250 /*
260 func bg_print(p;int,x;int,y;int,s;str)
270 int i:str a
280 for i=1 to strlen(s)
290   a=mid$(s,i,1)
300   bg_put(p,x,y,256+asc(a))
310   x=x+1
320 next
330 endfunc
```

リスト3

```
 10 /*
 20 /* スペースハリアーのフォントをかりる
 30 /*
 40 /*      Programed by H.Kageyama
 50 /*
 60 screen 1,3,1,1
 70 int i,j,cnt=0
 80 str wrtdat
 90 str mes1="sp={"
100 dim char font(63)
110 dim char num(43)={
120   32,33,48,49,50,51,52,53,54,55,
130   56,57,63,64,65,66,67,68,69,70,
140   71,72,73,74,75,76,77,78,79,80,
150   81,82,83,84,85,86,87,88,89,90,
160   46,44,45 }
170 ai=fopen("b:FONT.DAT","c")
180 get_font(0,1)
190 get_font(16,25)
200 get_font(31,60)
210 get_font(63,63)
220 fputc(&H1A,ai)
230 fcloseall()
240 end
250 /*
260 func get_font(s;int,e;int)
270 int i,j
280 for i=s to e
290   sp_pat(i,font,0)
300   fwrites(mes1,ai)
310   cr_lf()
320   wrtdat="+"
330   for j=0 to 55
340     wrtdat=wrtdat+str$(font(j))+","
350     if ((j+1) mod 8)=0 then {
360        fwrites(wrtdat,ai)
370        cr_lf()
380        wrtdat="+" }
390   next
400   for j=56 to 62
410     wrtdat=wrtdat+str$(font(j))+","
420   next
430   wrtdat=wrtdat+str$(font(63))+" }"
440   fwrites(wrtdat,ai)
450   cr_lf()
460   wrtdat="sp_def("+str$(num(cnt))+",sp,0)"
470   fwrites(wrtdat,ai)
480   cr_lf()
490   cnt=cnt+1
500   wrtdat="/*"
510   fwrites(wrtdat,ai)
520   cr_lf()
530   wrtdat="+"
540 next
550 endfunc
560 /*
570 func cr_lf()
580   fputc(&HD,ai)
590   fputc(&HA,ai)
600 endfunc
```

X68000用 ©SEGA
TURBO OUTRUNより RUSH A DIFFICULTY

Shindoh Noriyuki
進藤 慶到

X1/X1turbo用
パレードしようよ

Okada Kazuhiko
岡田 一彦

ずいぶんと暖かくなってきた今日この頃，新しいクラスにはもう慣れましたか？　さて，今回は3月号で予告したものの，ページの都合でひと月遅れになってしまったこの2曲をお送りします。遅くなってごめんなさい。それでは，しっかり打ち込んで堪能してください。

進藤君ですよ

う～む恐ろしい。なにが恐ろしいって，進藤君ですよ，進藤君。Oh!X LIVE in史上に残る名作とまで言われるメタルホークの彼ですよ。たった1曲で“進藤＝スゴイ”の方程式を作るなんざ，ちょっと信じられないことですぜ。このページだって，OPMAが発表されてからというもの，曲のレベルは上がり放題。特にゲームミュージックのレベルは，みんな甲乙つけがたいデキだったのに，なぜかずば抜けてすごい。編集室でも彼のイメージが強烈すぎて，この曲を採用する際に笑い話があったほど。

「え～また進藤君ですか，マズいんじゃないの～？」「ちょっと（曲数を）載せすぎですかね」「常連ですよね～」「何曲ぐらい載ったっけ？」「え～っと……1曲」「うっっそ～～！」ペラペラペラ（最近の資料を調べている）。「あっほんとだ」

その場になんともいえない空気がただよったのは言うまでもないでしょう。そういえばまだ曲を紹介してませんでしたね。曲は，TURBO OUTRUNよりRUSH A DIFFICULTYです。TURBO OUTRUNと言えば某FM TOWNSに移植され注目を浴びたSEGAの体感ゲームです。

さて肝心の作品のデキのほうですが，「進藤君」とひとこと言っておきましょう。SEGAの音源ドライバは，周知の如くバケモノなのですが，その分を差し引いて考える必要はありません。チャンネル数の関係でカウベルを入れられなかったと進藤君は言っていますが，ほとんど気にはならないでしょう。どうしてもカウベルが欲しい人は3月号に掲載されたOPMDを使って，手持ちのMIDI楽器のカウベルを使ってみてはいかがでしょうか。えっ楽器がない，失礼しました。

TURBO OUTRUN

なお，リスト入力上の注意点として，チャンネル番号の並び方に気をつけてください。このリストでは，ドラムを1チャンネル目に持ってきています。もちろん，ちゃんとした理由があります。OPMAでは後ろのトラックのほうが優先されます。このことを利用して，たとえばシンバル以外が同じリズムパターンならば，シンバルだけを後ろのトラックに持っていってしまえばドラムのチャンネルのデータ量の節約になるわけです。詳しくはリストを見てください。

初登場プリンセス・プリンプリン？

X1用にはPRINCESS PRINCESSのアルバムLOVERSより“パレードしようよ”です。PRINCESS PRINCESSといえばDIAMONDSで一躍有名になった女の子5人組のバンドです。DIAMONDSのほうも人気が高く，何作かが投稿されてきています。この曲はSONYのカセットテープのCFで使われていたので，聴いたことがあるかもしれませんね。

肝心の曲のデキのほうは原曲の明るさを見事に伝える，楽しい作りになっています。ただし，PSGの使い方が今ひとつのような気もしますので，もう少し研究してみてください。最近では1989年12月号のパズーとシータなどはうまかったと思います（ここだけの話ですが，最初はPSGが鳴っているなんて気づかなかったんです）。そのぐらいうまくまわりの音と溶け込ませることができれば，かなりのものでしょう。プログラムは3本に分かれています。1本目は音色データ，あとはオートロードになります。演奏にはMUSIC BASICと1989年5月号の拡張，1989年10月号の音色セットルーチンが必要です。音色セットルーチンは1990年3月号のねこバスのところにも載っていますので，1989年10月号を持っていない人はそちらのほうを入力してください。これからもこの音色定義は使われていくと思いますので，ぜひ入力してみてください。

PRINCESS PRINCESS

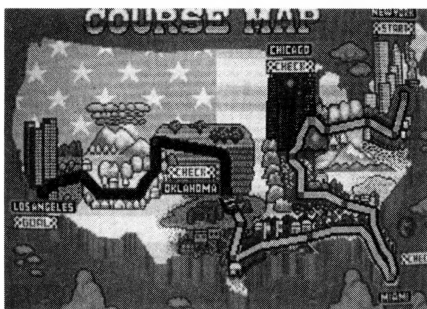
最後にLIVE質問箱

Q）MIDI MMLの立ち上げ方がまったくわかりません。BASIC V1.0を立ち上げて，NEWON & HB379として，セーブしたものをロードしてCALL & HA8B0としています。そしてバグを直してturbo用にCTCのアドレスを書き換えました。こうしたあとにTEMPO0とすると，みごとに暴走しました。入力ミスはないと思います。もし完成したらバンバン投稿するつもりです。

大阪府　森本和也

A）入力したもののバグを直してからセーブしてください。そしてNEWON & HB800としてCALL & HA8B0としてみてください。これでも動かないときは入力ミスか，MIDIボードのほうのミスです。ちなみにBASICはCZ8FB01のV1.0を使ってください。

(S.K.)

▶毎日毎日，会社で残業やって，真っ白に燃えつきた。俺はX68000の電源の入れ方も忘れてしまった。あ～かわいそうな，俺のX68000。吉田　貴（23）X68000ACE-HD　大阪府

リスト1 TURBO OUTRUN

```
  10 /*      save "RUSH!                .bas"
  20 /*
  30 /*    - T U R B O   O U T R U N -
  40 /*
  50 /*   R U S H   A   D I F F I C U L T Y
  60 /*
  70 /*    P R O G R A M E D   B Y   E N G
  80 /*
  90 m_init()
 100 key  3,"          @M
 110 key  9,"m_stop()@M
 120 key 10,"m_play()@M
 130 /*
 140 str p(30)[256]
 150 char o(255),v(4,9),voi(4,10)
 160 /*
 170 for i=1 to 8
 180  m_alloc(i,5000)
 190  m_assign(i,i)
 200 next
 210 /*
 220 VD()
 230 PD()
 240 m_play()
 250 end
 260 /*
 270 /*  S E T   M M L   T O   T R A C K
 280 /*
 290 func t(tt)
 300  r=0
 310  while o(r)<>255
 320   m_trk(tt,p(o(r)))
 330   r=r+1
 340  endwhile
 350 endfunc
 360 /*
 370 /*  V O I C E   S E T
 380 /*
 390 func set(vn)
 400  voi(0,0)=(v(4,1)*8)+v(4,0)
 410  voi(0,1)=15
 420  voi(0,9)=3
 430  for x=0 to 3
 440   for y=0 to 9
 450    voi(x+1,y)=v(x,y)
 460   next
 470  next
 480 m_vset(vn,voi)
 490 endfunc
 500 /*
 510 /*  V O I C E   D A T A
 520 /*
 530 func VD()
 540 /*
 550 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  BASS
 560 v={  31,  7,  7,  6,  2, 25,  3,  6,  0,  0,
 570      24,  6,  6,  3,  1, 54,  3,  5,  0,  0,
 580      31,  9,  7,  4,  1, 17,  2,  0,  0,  0, /* CON FBL
 590      31,  6,  6,  9, 15,  2,  2,  1,  0,  0,        0,  4}
 600 set(70)
 610 /*
 620 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  CHO1
 630 v={  31,  0,  0,  0,  0, 27,  1,  2,  0,  0,
 640      31,  0,  0,  0,  0, 30,  0,  2,  0,  0,
 650      31,  0,  0,  0,  0, 26,  1,  2,  0,  0, /* CON FBL
 660      31,  0,  0,  6,  0,  0,  0,  2,  0,  0,        3,  7}
 670 set(71)
 680 /*
 690 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  HIHAT
 700 v={  31,  0,  0,  5, 12, 18,  0, 15,  3,  1,
 710      31,  0,  0,  5, 10, 33,  0, 10,  3,  3,
 720      31,  0,  0,  5, 10, 15,  0, 15,  0,  3, /* CON FBL
 730      31, 17,  9,  8,  2,  0,  0,  1,  6,  0,        2,  7}
 740 set(72)
 750 /*
 760 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  HIHAT2
 770 v={  31, 31,  0,  0,  1,  2,  0, 11,  0,  3,
 780      31, 28,  2,  0,  1,  0,  2, 12,  0,  3,
 790      31, 22,  0,  1,  1,  5,  1,  1,  7,  1, /* CON FBL
 800      18, 18, 10,  9,  2,  0,  2,  7,  0,  0,        1,  5}
 810 set(73)
 820 /*
 830 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  ETOM
 840 v={  31,  4,  0,  0,  0,  3,  1,  8,  0,  2,
 850      31, 21,  9,  0,  6, 24,  1, 13,  3,  0,
 860      31, 26,  0, 15, 15,  0,  1,  4,  0,  1, /* CON FBL
 870      31, 12,  6, 15,  3,  0,  1,  4,  0,  0,        3,  7}
 880 set(74)
 890 /*
 900 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  PIANO
 910 v={  31,  0,  0,  0,  0, 32,  0,  4,  3,  0,
 920      28, 13,  9,  7,  2,  1,  1,  4,  3,  0,
 930      31,  0,  0,  0,  0, 35,  0, 12,  7,  0, /* CON FBL
 940      31, 13,  9,  7,  2,  1,  1,  4,  7,  0,        4,  7}
 950 set(75)
 960 /*
 970 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  MAIN
 980 v={  31,  0,  0,  9,  0, 45,  0,  3,  3,  0,
 990      31, 19,  9,  9,  1,  5,  0,  2,  7,  0,
1000      31,  0,  0,  9,  0, 43,  0, 14,  7,  0, /* CON FBL
1010      31, 19,  9,  9,  1,  0,  0,  4,  3,  0,        4,  7}
1020 set(76)
1030 /*
1040 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  PIANO
1050 v={  31,  0,  0,  0,  0, 32,  0,  8,  3,  0,
```

```
1060      31, 14,  8,  6,  1,  0,  0,  8,  3,  0,
1070      31,  0,  0,  0,  0, 24,  0,  4,  7,  0, /* CON FBL
1080      31, 14,  8,  6,  1,  0,  0,  4,  7,  0,        4,  7}
1090 set(77)
1100 /*
1110 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  MAIN
1120 v={  31,  0,  0,  0,  0, 45,  0,  3,  3,  0,
1130      31, 19,  9,  6,  1,  5,  0,  2,  7,  0,
1140      31,  0,  0,  0,  0, 43,  0, 14,  7,  0, /* CON FBL
1150      31, 19,  9,  6,  1,  0,  0,  4,  3,  0,        4,  7}
1160 set(78)
1170 /*
1180 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  CHORD
1190 v={  31,  0,  0,  0,  0, 24,  0,  8,  7,  0,
1200      31,  0,  0,  6,  1,  0,  0,  8,  3,  0,
1210      31,  0,  0,  0,  0, 18,  0,  4,  3,  0, /* CON FBL
1220      31,  0,  0,  6,  1,  9,  0, 12,  7,  0,        4,  7}
1230 set(79)
1240 /*
1250 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  CHO1
1260 v={  31,  0,  0,  0,  0, 20,  0,  8,  3,  0,
1270      21, 14, 10,  9,  1,  0,  0,  8,  3,  0,
1280      31,  0,  0,  0,  0, 23,  0,  8,  7,  0, /* CON FBL
1290      21, 14, 10,  9,  1,  7,  0,  8,  7,  0,        4,  6}
1300 set(80)
1310 /*
1320 /*   AR 1DR 2DR  RR 1DL  TL  RS MUL DT1 DT2  PBRS
1330 v={  16, 16,  0,  3,  3, 26,  0,  3,  3,  0,
1340      19,  0,  0,  6,  0, 27,  0,  1,  0,  0,
1350      26,  0,  0,  6,  0, 26,  0,  1,  0,  0, /* CON FBL
1360      27,  0,  0, 11,  0, 10,  0,  1,  0,  0,        0,  7}
1370 set(81)
1380 /*
1390 endfunc
1400 /*
1410 /*  P L A Y   D A T A
1420 /*
1430 func PD()
1440 /*
1450 p(0)="@70o2@v127q8p3l16y49,28
1460 p(1)="f8<c8>a-8b-f8a-gfe-ce-8 f8<c8>a-8b-f8fa-gffe-c
1470 p(2)="f8<c8>a-8b-f8a-gfe-ce-8 f8<c8>a-8b-f8fe-fb-<ce-f>
1480 p(3)="f8<c8>a-8b-f8a-gfe-ce-8 f8.f8<cf>fe-8.e-8<gfe->
1490 p(4)="o2|:f8<c8>a-8b-f8a-gfe-ce-8:|"+p(2)
1500 p(5)="|:o3d-8d-d-d-8d-d-8d-8.d-d-8d- d-8d-d-8d-8d-8d-8d-8d
-c8> b-8b-b-b-8b-b-8b-8b-b-fa-b- b-8b-b-8b-8b-8b-8b-a-b-:|
1510 p(6)=  "o3d-8d-d-d-8d-d-8d-8.d-d-8d- d-8d-d-8d-8d-8d-8d-8d
-c8> b-8b-b-b-8b-b-8b-8b-b-fa-b- <c<c>c>b<b>bb-<b->b-a<a>a<a->a-
<g>g
1520 p(7)="@71o1v11l1f&f&f&f&f&f&f l16@70o2@v127f8.f8<d-fc>e-8.
e-8<gfe-
1530 p(28)="[d.c.]r4.
1540 p(29)="[coda]
1550 p(30)="[*]
1560 o={28,29,0,1,2,1,3,4,4,4,4,0,1,2,1,3,4,4,4,4,5,6,7,4,4,4,4
,5,6,30,255}
1570 t(2)
1580 /*
1590 /*
1600 p(0)="@71o4v12q8p3l16y50,12
1610 p(1)="|:3v13p3e-4v9rv13e-v9rv13e-v9p1r2r8.v13p3e-v9rv13e-v
9p1r8r8.v13p3dv9rv13dv9p1r8:|p3v13e-4v9rv13e-v9rv13e-v9r2v13e-8.
e-v9r4v13d8.dv9r4
1620 p(2)="@v0@77o2p1 |:3rrv9a-rra-rra-rra-rrga- rra-rra-rrra-r
a-a-ra-r:||:rrv9@77p1a-rra-rrr@80v11b-b-b-b-b-b-:|
1630 p(3)="v12o2p1|:5rrffffrrffffre-rf rrffffrrrffffff:|
1640 p(4)="rrffffrrrffffff v11l8c.>b.b-.a.l16a-rgr
1650 o={28,29,0,1,2,2,0,1,2,2,3,4,0,1,2,2,3,4,30,255}
1660 t(3)
1670 /*
1680 /*
1690 p(0)="@71o4v12q8p3l16y51,48
1700 p(1)="|:3v13p3c4v9rv13cv9rv13cv9p2r2r8.v13p3cv9rv13cv9p2r8
r8.>v13p3b-v9rv13b-v9p2r8<:|p3v13c4v9rv13cv9rv13cv9r2v13c8.cv9r4
>v13b-8.b-v9r4
1710 p(2)="@v0@77o2p2 |:3rrv9frrfrrfrrfrre-f rrfrrfrrrfrffrfr:|
|:rrv9@77p2frrfrrr@80v11cccccc:|
1720 p(3)="@v0@71o4p3|:5rrv12p3a-a-a-a-v8p2rrv12p3a-a-a-a-v8p2r
v12p3a-v8p2rv12p3a- v8p2rrv12p3a-a-a-a-v8p2rrrv12p3a-a-a-a-a-a-a
-v8p2:|
1730 p(4)="rrv12p3a-a-a-a-v8p2rrrv12p3a-a-a-a-a-a-a-<v10l8c.>b.
b-.a.l16a-p1rp3gp1r
1740 t(4)
1750 /*
1760 /*
1770 p(0)="@71o3v12q8p3l16y52,24
1780 p(1)="|:3v13a-4v9rv13a-v9rv13a-v9r2r8.v13a-v9rv13a-v9r8r8.
v13gv9rv13gv9r8:|v13a-4v9rv13a-v9rv13a-v9r2v13a-8.a-v9r4v13g8.gv
9r4
1790 p(2)="@v0@77o2p3 |:3rrv9crrcrrcrrcrr>b<c rrcrrcrrrcrccrcr:
||:rrv9@77crrcrrr@80v11ffffff:|
1800 p(3)="@v0@71o5p3|:5rrv11p3ccccv8p2rrv11p3ccccv8p2rv11p3cv8
p2rv11p3c v8p2rrv11p3ccccv8p2rrrv11p3cccccccv8p2:|
1810 p(4)="rrv11p3ccccv8p2rrrv11p3ccccccc v10l8c.>b.b-.a.l16a-p
2rp3gp1r
1820 t(5)
1830 /*
1840 /*
1850 p(0)="@79o2v13q8p3l16y53,20
1860 p(1)="l24|:28e-&c&f&e-&c&>b-<:|l16v12<fca-fca-fce->b-<ge->
b-<ge->b-
1870 p(2)="@79o1v10l1|:f&fg&ge-&e-|1c&c:||2<v12c&c
1880 p(3)="v11c&c>a-&a-<e-&e-c&c>v12a-&a-&a-l16p2v11<c&<c&>c&>b
&<b&>b&b-&<b-&>b-&a&<a&>a&<a-&>a-&g&<g
1890 p(4)="@79o1v10l1p1 f&f&f&fg&ga-2b-2<c
```


▶白地に黒のタイトル？　きっと白地になったのはX1（マニアタイプ）のスノーホワイトの怨念かな？
末吉　宏（20）X1turbo　兵庫県

```
1900 o={28,29,0,1,2,0,1,2,3,4,2,3,30,255}
1910 t(6)
1920 /*
1930 /*
1940 p(0)="@75o2v12q8p3l16y54,16
1950 p(1)="y2,5|:3c8f8e-8c8fe-rc8f8e- c8f8e-8ca-8e-rcg8e-8:|c8f
8e-8c8fe-rc8f8e- v12fca-fca-fcge->b-<ge->b-rr
1960 p(2)="y2,5@76v14o5g32&a-16.g8fe-rfrc2..>b-8.a-b-r<e-rc&c2.
..>>v11e-fga-b-<v12cd-e-fv13ga-b-<cd-e-v14fg
1970 p(3)="g32&a-.g8fe-8frc2..g32&a-.&a-b-8<ccr>frf2...&f1
1980 p(4)="y2,5|:o5c8cc8>a-rf8.&f2e-8.f8.g8.a-8.b-8:|o5c8>b-8a-
8b-f8f4.<l8c.>b-a-b-<c>b-<e-d-
1990 p(5)="cc16c8>{a-r}f.&f2a-4gfe-ce-
2000 p(6)="y2,5l16f>fga-b-<cd-e- f>a-b-<cd-e-fg a-cd-e-fga-b- <
c>e-fga-b-<cd-
2010 p(7)="y2,5e->fga-b-<cd-e-f>ga-b-<c>b-<e-d-@81o5v15c&c&>c&>
b&<b&>b&b-&<b-&>b-&a&<a&>a&<a-&>a-&g&<g
2020 p(8)="@79o1v10l1p2 f&f2g2a-&a-2b-2<c&c2l2de-fga-
2030 p(9)="y2,5@76v14o4l16fga-fga-b-ga-b-<c>a-b-<cd>b- <cde-cde
-fde-fge-f@78{ge-}@76fg
2040 p(10)="|:3a-gfe-c:|b-a-gfe-dc>b-a-ga-b-<c>b-a-ge-
2050 p(11)="fa-b-<c>b-a-fa-b-<c>b-a-f@78{a-b-}@76<c>b- a-fe-fa-
b-<ce-fe-c>b-<c>b-a-f
2060 p(12)="e-fa-f{e-fa-}8b-<cfga-gb-a-{fe-c}8efe-c>b-a-fe-ce-f
a-b-<ce-f
2070 p(13)="y2,5r|:10fe-c:||:5ge-c:||:3a-fc:|b-a-gfe-efe-c
2080 p(14)="rde-dc>b-a-ga-b-<c>b-a-gfe-fgfe-ce-fga-ga-b-b<c>b-a
- ga-gf8@78q6<c8ccf8ffa-8a-a-<c8cc>a-8gga-gf&f4@76q8
2090 p(15)="y2,5e-d-cd-e->ff8<e-d-cd-e->fff f<e-d-c>b-a-gfe-fgf
e-ce-c
2100 p(16)=">b-<ce-ce-fgfga-b-a-b-b<c>b-a-gfe-fe-ce-c>b-<c>b-a-
b-a-g
2110 p(17)="fe-fa-b-<c>b-a|:fa-b-<c>b-a-fe-:|ce-fa-b-<ce-f
2120 p(18)="rga-gfa-gfb-a-gfga-fe-e-ce-cfe-ce-cfe-cfe-ce-
2130 p(19)="|:ccc>b-a-f<:|cc>b-a-fa-b-<ce-ce-effe-ce-c>b-a- y2,
5<e-c>b-a-b-<ce-ce-fgfga-b-b
2140 p(20)="@81o5v15c&c&>c&>b&<b&>b&b-&<b-&>b-&a&<a&>a&<a-&>a-&
g&<g
2150 p(27)="@v127l64o1d&d-&c&>b&b-&a&a-&g
2160 p(28)="[d.c.]@72o4v12q5p1l16y54,00|:4g16:|@74p1"+p(27)
2170 o={28,29,0,1,2,3,2,3,0,1,2,3,2,3,
2180     4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,30,255}
2190 t(7)
2200 /*
2210 /*
2220 p(0)="@72o4@v127q8p1l16y55,48
2230 p(1)="|:28q2p3grq8p1g8:|
2240 p(2)="@76o4v13p1fca-fca-fcge->b-<ge->b-rr@v0@72o4@v127
2250 p(3)="|:64q2p3grq8p1g8:|
2260 p(4)="|:44q2p3grq8p1g8:|p1<b4.b4.b4>
2270 p(5)="p1q2|:64@v127cv12c:|@v127
2280 p(28)="[d.c.]@72o4v13q5p2l16y55,24|:4g16:|@74p2"+p(27)
2290 o={28,29,0,1,2,3,1,2,3,4,5,3,4,30,255}
2300 t(8)
2310 /*
2320 /*
2330 p(0)="@73o5@v126q3p2l16y48,20 y3,3
2340 p(1)="y2,23rcy2,14rcy2,23cry2,14rc y2,23rry2,14q8cy2,23rq3
cy2,23ry2,14rq6cq3
2350 p(2)="y2,23ccy2,14rcy2,23cry2,14ry2,23c ry2,23ry2,14cry2,2
3rcy2,14ry2,15r
2360 p(3)="y2,5rrry2,23rl24y2,14ry2,15ry2,15ry2,12ry3,1y2,12ry3
,2y2,13r l16y3,3y2,5rccy2,23rccy2,14rr
2370 p(4)="|:3"+p(1)+p(2)+":|"+p(1)
2380 p(5)="y2,23ccy2,14rcy2,23cy3,1y2,11ry3,3y2,14ry3,1y2,11cy2
,11ry3,3y2,12ry2,14cy2,12ry2,12ry3,2y2,13cy3,3y2,14ry3,2y2,13ry3
,3
2390 p(6)="y2,23rcy2,14ry2,23cy2,23cry2,14rc y2,23rry2,14q8cy2,
23rq3cy2,23ry2,14rq6cq3
2400 p(7)=p(6)+"|:3"+p(2)+p(1)+":|
2410 p(8)=p(6)+p(2)
2420 p(9)="y2,23rcy2,15ry2,15cy2,23cy2,15cy2,15ry2,16c y2,15ry3
,1y2,12c32y3,2y2,12r32y3,3y2,13c32y2,13r32y2,23ry2,13ry2,13ry2,2
2rr
2430 p(10)="y2,5rcy2,15cy2,23ry2,6ry2,15ry3,1y2,5rcy3,3y2,15cy2
,23rry2,15ry3,2y2,5rcy3,3y2,14cc
2440 p(11)="y2,5rrrr|:6y2,3r:|y2,3cy2,3v13c@v127|:3y2,3r:||:3y2
,2r48:|
2450 p(12)="y2,14r32y2,14r32rcc|:4y2,16r:||:5y2,15r:||:3y2,14r:
|
2460 p(13)="ry2,23rry2,15rcy2,12ry2,13rr y2,15cy2,16ry2,12r24y3
,1y2,12c24y3,2y2,13r24y3,3y2,23ry2,14ry2,15l48ry2,15ry2,12cy3,1y
2,12ry3,2y2,13ry3,3y2,23c l16
2470 p(14)="y2,23rccy2,14rry2,14ry2,2r24y2,2c24y2,2r24 y2,23ccr
y2,14crcy2,14ry2,15r
2480 p(15)="ry2,23ry2,23ry3,2y2,13ry3,3y2,13ry3,1y2,12ry2,12ry2
,12r
2490 p(16)="y3,3y2,2cy3,2y2,8ry3,3y2,2cy2,6cy3,2y2,8ry3,3y2,2cy
2,6cy2,8r y3,3y2,14rry2,14ry2,5rcy2,5cry2,12r y2,13ry2,14cy2,23r
y2,23ry2,14cy2,23ry2,23cy2,14r
2500 p(17)="y2,23cry2,23cry2,14ry2,15cry2,23ry2,23cy2,16r32y2,1
5r32y2,15ry2,15c|:4y2,14q7c:|q3
2510 p(18)="y2,5q8c&cy2,14cy2,23cry2,16r32y2,15r32y2,14ry2,23r
y2,5c&c&cy2,14c|:3y2,23r24:|y2,14rrq3
2520 p(28)="[d.c.] t123 y15,0 y3,3|:4y2,6r16:|y2,14r8
2530 o={28,29, 0, 4, 3, 4, 5, 4, 5, 0, 4, 3, 4, 5, 4, 5,
2540      0, 7, 5, 8, 9,10,11,12,13,14,15,16,17,18,
2550      4, 5, 4, 5, 0, 7, 5, 8, 9,10,30,255}
2560 t(1)
2570 /*    注意：チャンネル番号が順番に並んでいませんので、
2580 /*          よく見て入力して下さい。ドラムがトラック１です
。
2590 endfunc
2600 /*
```

日本音楽著作権協会(出)許諾第8972850-901号

リスト2 パレードしようよ 1

```
10 '*******************************************
20 '*     L O V E R S     ﾊﾟﾚｰﾄﾞ ｼｮｳﾖ          *
30 '*                                         *
40 '*            PRINCESS PRINCESS            *
50 '*                                         *
60 '*          PROGRAMED BY KAZUHIKO OKADA    *
70 '*                                         *
80 '*******************************************
90  DV$=MEM$(&H7498,2):IF RIGHT$(DV$,1)=":" AND LEFT$(DV$,1)<>CH
R$(0) THEN DEVICE DV$
100 A=PEEK(&HB000):IF A<>&HFE THEN LOADM"0:VOICE SETTER.Rou"
110 DEFSTR A-Z:DEFINT i,j,N,V:CLEAR&HFF00
120 DEFUSR=&HB000:DIM V(4,10):DIM p(20)
130 DEFFNV$(N,V(0,0))=USR(CHR$(N)+MKI$(VARPTR(V(0,0))))
140 GOTO190
150 LABEL"READ"
160 FOR J=0 TO 10:FOR I=0TO 4:READ V(I,J):NEXT:NEXT
170 RETURN
180 '######## VOICE #######
190 'SOUND NUMBER 1      Bass
200 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
210 DATA  58, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
220 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
230 DATA  31,  8,  0,  6,  4, 34,  0,  1,  7,  0,  0
240 DATA  31, 16,  0, 14,  5, 45,  1,  3,  6,  0,  0
250 DATA  31,  5,  0, 12,  4, 28,  0,  1,  2,  0,  0
260 DATA  28,  7,  0,  7,  3,  0,  0,  1,  3,  0,  0
270 "READ":A=FNV$(1,V)
280 'SOUND NUMBER 2      E.Guitar
290 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
300 DATA  40, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
310 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
320 DATA  31,  2,  1,  0,  1, 13,  0,  4,  1,  0,  0
330 DATA  27,  1,  1,  4,  1, 37,  0, 10,  0,  0,  0
340 DATA  31,  2,  1,  0,  3, 29,  0,  0,  0,  0,  0
350 DATA  31, 12,  1,  7,  1,  0,  0,  1,  0,  0,  0
360 "READ":A=FNV$(2,V)
370 'SOUND NUMBER 3      brass
380 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
390 DATA  60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
400 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
410 DATA  16,  8,  5,  5,  2, 27,  0,  4,  3,  0,  0
420 DATA  18,  0,  0,  7,  0,  0,  0,  4,  3,  0,  0
430 DATA  10,  9,  5,  5,  2, 27,  0,  4,  4,  0,  0
440 DATA  13,  0,  0,  6,  0,  0,  0,  4,  4,  0,  0
450 "READ":A=FNV$(3,V)
460 'SOUND NUMBER 4      strings
470 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
480 DATA  60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
490 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
500 DATA  31,  0,  0,  0,  0, 27,  0,  8,  3,  0,  0
510 DATA  12,  0,  0,  7,  0,  0,  0,  8,  3,  0,  0
520 DATA  31,  0,  0,  0,  0, 20,  0,  8,  7,  0,  0
530 DATA  10,  0,  0,  7,  0,  0,  0,  8,  7,  0,  0
540 "READ":A=FNV$(4,V)
550 'SOUND NUMBER 5      Acorstic Piano
560 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
570 DATA  42, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
580 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
590 DATA  28,  0,  1,  0,  1, 35,  1,  1,  0,  0,  0
600 DATA  27,  1,  1,  1,  7, 47,  1,  4,  0,  0,  0
610 DATA  28,  1,  0,  1,  8, 48,  1,  8,  0,  0,  0
620 DATA  27,  7,  4,  5,  2,  0,  2,  2,  0,  0,  0
630 "READ":A=FNV$(5,V)
640 'SOUND NUMBER 6      B.Drum
650 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
660 DATA  62, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
670 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
680 DATA  31, 24,  0, 10, 11,  2,  0,  0,  0,  0,  0
690 DATA  31, 15, 16,  9,  4,  0,  0,  0,  0,  0,  0
700 DATA  31, 15, 16,  9,  4,  3,  0,  0,  0,  0,  0
710 DATA  31, 15, 16,  9,  4,  0,  0,  1,  0,  0,  0
720 "READ":A=FNV$(6,V)
730 'SOUND NUMBER 7      S.Drum
740 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
750 DATA  60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
760 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
770 DATA  31,  0,  0,  1,  0,  0,  0, 15,  0,  3,  0
780 DATA  31, 15, 11,  7,  4,  0,  0,  2,  0,  2,  0
790 DATA  31, 24,  0,  9, 15,  0,  0,  2,  0,  2,  0
800 DATA  31, 17, 15,  8,  2,  0,  0,  1,  0,  0,  0
810 "READ":A=FNV$(7,V)
820 'SOUND NUMBER 8      tom tom
830 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
840 DATA  59, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
850 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
860 DATA  31,  0,  0,  0,  0,  0,  0, 10,  0,  3,  0
870 DATA  31, 27, 17,  1,  6, 17,  0, 13,  0,  0,  0
880 DATA  31, 29,  0, 15, 15,  0,  0,  0,  0,  0,  0
890 DATA  31,  6,  0,  8, 15,  0,  0,  1,  0,  0,  0
900 "READ":A=FNV$(8,V)
910 'SOUND NUMBER 9      Kaori !!
920 '      AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
930 DATA  60, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
940 '      AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
950 DATA  31,  0,  0,  0,  0, 27,  0,  2,  3,  0,  0
960 DATA  31,  7,  0,  6,  2,  0,  0,  2,  7,  0,  0
970 DATA  31, 18, 10,  6,  7, 25,  0,  3,  3,  0,  0
980 DATA  31, 12, 10,  7,  2,  0,  0,  1,  7,  0,  0
990 "READ":A=FNV$(9,V)
```

▶ぅっ。「道楽絵四」とは「ドラクエⅣ」だったのか。４月号のP132を見るまで気づかなかった。私も落ちたものよ(受験も落ちた)。　寺門　修司 (18) X1turboZ　兵庫県

```
1000 'SOUND NUMBER 10    Kaori 2 !!
1010 '       AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
1020 DATA  59, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
1030 '       AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
1040 DATA  27,  9,  3,  4, 10, 47,  3,  6,  3,  0,  0
1050 DATA  26,  7,  0,  6,  5, 45,  2,  4,  0,  0,  0
1060 DATA  24,  8,  0,  6,  1, 38,  2,  2,  6,  0,  0
1070 DATA  19,  6,  4,  9,  1,  0,  0,  2,  1,  0,  0
1080 "READ":A=FNV$(10,V)
1090 'SOUND NUMBER 11    E.Guitar 2
1100 '       AF  OM  WF  SY  SP PMD AMD PMS AMS PAN /
1110 DATA  40, 15,  0,  0,  0,  0,  0,  0,  0,  3,  0
1120 '       AR  DR  SR  RR  SL  OL  KS  ML DT1 DT2 AM-EN
1130 DATA  31, 17,  1,  0,  7, 10,  1,  3,  4,  0,  0
1140 DATA  18,  1,  1,  8,  1, 28,  0, 15,  4,  0,  0
1150 DATA  31,  9,  1,  0,  1, 20,  3,  7,  7,  0,  0
1160 DATA  31, 12,  2,  8,  1,  0,  1,  1,  7,  0,  0
1170 "READ":A=FNV$(11,V)
1180 RUN"0:ﾊﾟﾚｰﾄﾞ ｼｮｳﾖ 2.mml
```

リスト3　パレードしようよ　2.mml

```
10 '        PROGRAM 2   (Ch1-Ch7)
20 DEFSTR a-z:DEFINT i:DIM p(25):CLS0:TEMPO0
30 PLAY "t178";
40 '------------- bass -----------
50 up="s3,1,0,12=3
60 dn="s4,1,0,12=3
70 p(0)="i1q8l12v14k5o2   =0
80 p(1)="d1&d2r6<b>c6d&d1&d2.<f4&f1&f2&f6>d+e6f&f1&f2f6ee-6d&d1&
d2&d6<b>c6d&d1 r4d4r4d4
90 p(2)=STRING$(2,"l4q7gf+ed<b>cdgcdegdef+a")
100 p(3)="v15q6<ggggggg>d ggggggg6d12"+dn+"e=0 <g6g12ggggggg >gg
gggggg <ggggggg>"+up+"d=0
110 p(4)="ggggggg6d12"+dn+"e=0 <g6g12gggggg"+up+"g=0 >ggggggf+f+
120 p(5)=STRING$(2,"ee<b>"+dn+"e=0c+c+<a6>c12<a>")+"<ef+gb>c+<a>
c+<a>ddd"+up+"a=0l12>"+STRING$(12,"d")+"l4<
130 p(6)="<aaa6a12abbb6g12b >ccc6<g12>c c+c+c+6<a12>c c<b>cc+
140 p(7)="dddd v13<g1&g2r8>"+up+"d4.=0 g1&g2r8=3g4.=0>g1&g1&g6g1
2d6<g12&"+dn+"g4.=0<b8&b>cc+d
150 p(8)="dd<a6>d12<f+ g.>e-6&e-4.<a-12&a-4.>d-8&d-2
160 p(9)="l12d1&d2r6<b>c6d&d1&d2.<f4&f1&f2&f6>d+e6f&f1&f2f6ee-6d
&
170 p(10)=LEFT$(p(9),LEN(p(9))-3)+"4
180 p(11)="l12 d6dr2d6d r4r6"+dn+"a=0r2 d6dr2d6d l6rdd+eff+
190 p(12)="l12>"+STRING$(12,"d")+"<
200 p(13)="t170 e-.<a-8&a->d-& d-8<g-.b t155e t60 g1
210 FOR i=0 TO 5:PLAY p(i);:NEXT:PLAY p(2);
220 FOR i=6 TO 7:PLAY p(i);:NEXT
230 FOR i=3 TO 5:PLAY p(i);:NEXT:PLAY p(2);PLAY p(6);
240 FOR i=8 TO 11:PLAY p(i);:NEXT:PLAY p(2);PLAY p(6);
250 PLAY p(12);:PLAY p(2);:PLAY p(13);
260 FOR i=0 TO 20:p(i)="":NEXT
270 PLAY ":";
280 '------------- vocal & etc. ----------
290 p(0)="i3q8l12v14k2o3=0
300 A="d1&d2r6<b>c6d&d1&d2.f4&f1&f2r6d+e6f"
310 p(1)=A+"&f1&f2f6ee-6d&d1&d2r6<b>c6d&d2.r4d4d4c4d4
320 p(2)=STRING$(6,"r1")+"p3r2.r6g&g4f+4e8f+4.   i9v13k2o3p3q8l1
2
330 p(3)="b6>cd6g&g4<b6>c d6g&g6rg6ga6g e4.c+8&c+4a4& a2r2 <a6b>
c6g&g4<a6b
340 p(4)=">c6g&g6rg6ga6g b4a4g6de6d& d2r2 <b6>cd6g&g4<b6>c d6g&g
6rg6ga6g
350 p(5)="e4.c+4.a4& a4r4r6bb6>c& c4<b4g6e4g& g4e6eg4a4 a4g6g&g2
 r1
360 p(6)="i10 r4g4e6f+4g&g2r2 g6gg6ge6f+4g&g2r2 g4g4e6f+4g&g4¯2e
4¯1g4¯b4¯ b4.a8&a4b4&b8a4.r4_5 i9¯3d6¯3b&
370 p(7)="b2r4b6>c& c6c<b6ba6g4g& g2r4d6b& b4a6ag6a4b& b2r6bb6b
>c4<b6ba6g4g&
380 p(8)="g2r6dd6b& b4a4g6ar4 _3g6gg6ge6f+4g& g4e4g6b4b& b4a4g6a
4e& e2r6ef+6g& g4>c4<b6e4g
390 p(9)="b4a4g6a4g&g1_3
400 p(10)="r1r1r1 i3o3v14 r2.r6f& f6de6fr6de6f r2.r6b& b4>c4c+4d
4 i9v13o3q8
410 p(11)="b4¯3a6ag6a4g&g1&g2r2_3 i3l12v15o3
420 p(13)=p(3)
430 p(14)=">c6g&g6rg6ga6g b4a4g6de6d& d2r2 <b6>cd6g&g4<b6>c d6g&
g6 g g6ga6g
440 p(15)="e4.c+4.a4& a4r4r6bb6>c& c4<b4g6e4g& g4e6eg4a4 a6g g6g
&g2 r1
450 p(16)="i10 r4g4e6f+4g&g2r2 g6gg6ge6f+4g&g2r2 g6gg6g e6f+4g&g
4¯2e4¯1g4¯b4¯ b4.a8&a4b4&b8a4.r4_5 i9¯3d6¯3b&
460 p(17)=p(7)
470 p(18)="g2r6dd6b& b4a4g6ar4 _3r6rg6&g e6f+4g& g4e4g6b4b& b6ba
6a g6a4e& e2r6ef+6g& g4>c4<b6e4b&
480 p(19)=A+"&f1&f2f6ee-6d&"+A+"r1r6e-f6a>c4r4< v14
490 p(20)=">d6dr2d6dr1d6dr2d6d l6<rdd+eff+    i9v15l12o4
500 p(21)="b4a4g4a6g&g1"+STRING$(7,"r1")+"i3o3l4v15 g.g.f.f.d+d+
d1
510 FOR i=0 TO 10:PLAY p(i);:NEXT
520 FOR i=13 TO 18:PLAY p(i);:NEXT:PLAY p(11);
530 FOR i=19 TO 20:PLAY p(i);:NEXT
540 PLAY p(7)+p(8);:PLAY p(21);
550 PLAY ":";
560 '
570 p(0)="i3q8l12v11k8o3=0 r16
580 p(1)=A+"&f1&f2f6ee-6d&d1&d2r6<b>c6d&d2.r8.v14p1<b4a4g4a4>
590 p(2)=STRING$(6,"r1")+"r2.r6d&d4d4c8d4.    i9 v11k8o3p3q8l12r
16
600 p(10)="r1r1r1 i3o3v12 r2.r6f& f6de6fr6de6f r2r8.r6v14g& g4a4
a+4b4 i9v11o3r16
610 p(11)="b4¯3a6ag6a4g&g1&g2r4.r16_3 i3l12v15o3
620 p(19)=STRING$(9,"r1")+"r6<b>c6ef+4g6a r1 b6b-a6a-g6ef4 r2e-6
de-6g f4e-6de-6gf4 r1 r16r6v12e-f6a>c4<r8.v14
630 p(20)="a6ar2a6ar1a6ar2a6a r16v12l6<rdd+eff+ i9v13l12o4
640 p(21)="b4a4g4a6g&g1r2.r8."+STRING$(6,"r1")+"i4o1L4v14q6e-.e-
.d-.d-.<bb>q8a1
650 FOR i=0 TO 10:PLAY p(i);:NEXT
660 FOR i=13 TO 18:PLAY p(i);:NEXT:PLAY p(11);
670 FOR i=19 TO 20:PLAY p(i);:NEXT
680 PLAY p(7)+p(8);:PLAY p(21);
690 FOR i=0 TO 20:p(i)="":NEXT
700 PLAY ":";
710 '------------- chord 1 -----------
720 p(0)="i4q8l1  v10 K5o2=0   p3
730 p(1)="d&d&d&d2.f4&f&f&f&f d&d&d i3p2v14o2g4f+4e4f+4
740 p(2)=STRING$(6,"r1")+"l12  r2.r6b&b4a4g8a4.
750 p(3)="i2o3v10p3g8<r4.r6ab6>d r4r6<dd8g4. r2r6ab6>e r4r6<dd8g
4. r2r6ga6>c
760 p(4)="r4r6<de8g4. r2r6ab6>d r6>gr6rg8g4.<r2r6<ab6>d r4r6<de8
g4.
770 p(5)="r2r6ab6>e r4r6<de8g4. r2r6ga6>c r4r6<de8g4. r2r6ab6>d
>r6gr6rg8g4.<
780 p(6)="i4p2o0v9r1r2b2&b1&b4¯4a4¯2g4¯ a4_8b1&b1 i3p3o3v13b4.a8
&a4b4&b6a&a2.
790 p(7)="i4o2l1v11 d&d&d&d&d&d&d&d <v11p1gggbgb>
800 p(8)="i1lp3v13o2l12 r6gr4g6gr4 r6de6f&f6fe6d r6gr4g6gr4 r4a9
6&a+96&b.&b48>d&d6d<a6g r6gr4g6gr4 r6de6f&f6fe6d r6gr4g6gr6<b& b
4>c4c+4d4
810 p(9)="i4p3o0v13l12 b6>dg6e-&e-6gb6c& c6e-a-6f&f6a->c6d&l1
820 A="d&d&d&d2.f4&f&f&f
830 p(10)=A+"&f2f6e12e-6d12&"+A+"&f
840 p(11)="v13o2l12 d6dr2d6d r1 d6dr2d6d l6rdd+eff+
850 p(12)="p3v11 d&d&d&d&d&d&d&d  <<l4v13q6 b-.>c.<a-.b-.fg+q8f+
1
860 FOR i=0 TO 8:PLAY p(i);:NEXT
870 FOR i=3 TO 7:PLAY p(i);:NEXT
880 FOR i=9 TO 11:PLAY p(i);:NEXT
890 PLAY p(7)+p(12);
900 FOR i=0 TO 20:p(i)="":NEXT
910 PLAY ":";
920 '--------- E.guitar ------------
930 p(0)="i2q8l12v12k04o3=0 p3
940 A="d1&d2&d6<b>c6d&d1&d2.<f4&f1&f2&f6>d+e6f&f1&f2f6e"
950 p(1)=A+"e-6d&d1&d2&d6<b>c6d&d1&d1>v11
960 p(2)=STRING$(2,"r6dr6<g>d6dr4 <r6dg6ar6dg6g r6>cr6<g>c6cr4 r
4d6rr8d4.")
970 p(3)="v10o3b8r4.r6ab6>d r4r6<dd8g4. r2r6ab6>e r4r6<dd8g4. r2
r6ga6>c
980 p(4)="r4r6<de8g4. r2r6ab6>d r6br6e>c8<b4. r2r6<ab6>d r4r6<de
8g4.
990 p(5)="r2r6ab6>e r4r6<de8g4. r2r6ga6>c r4r6<de8g4. r2r6ab6>d
r6br6f+>c8<b4.
1000 p(6)="<e1c+1e1c+1 e1c+2<a2> i3o3v13f+4.f+8&f+4f+4&f+6f+&f+2
. i2o4v12
1010 p(7)="i4p2o1v11l1 cdeeef+
1020 p(8)="i2p3v12o3l12 r6gr4g6gr4 r6de6f&f6fe6d r6gr4g6gr4 r4a9
6&a+96&b.&b48>d&d6d<a6g r6gr4g6gr4 r6de6f&f6fe6d r6gr4g6gr6<b& b
4>c4c+4d4
1030 p(9)="i2p3l12v13o3 d4&d6b-&b-4&b-6e-&e-4&e-6a-&a-2
1040 p(10)=p(0)+a+"e-6d&"+A+"e-4
1050 p(11)="i2l12v12o3 d6dr2d6d r1 d6dr2d6d l6rdd+eff+ o4v12l12
1060 p(12)="i2l4 v13o3q7 e-.<a-.>d-.<g-.be q8g1
1070 FOR i=0 TO 6:PLAY p(i);:NEXT:PLAY p(2);
1080 FOR i=7 TO 8:PLAY p(i);:NEXT
1090 FOR i=3 TO 6:PLAY p(i);:NEXT:PLAY p(2);:PLAY p(7);
1100 FOR i=9 TO 11:PLAY p(i);:NEXT
1110 PLAY p(2)+p(7);:PLAY "i2o4v11l12p3"+p(2)+p(12);
1120 FOR i=0 TO 20:p(i)="":NEXT
1130 PLAY ":";
1140 '------------- chord   -----------
1150 p(0)="i5q8l4 v14k5o4=0 p3
1160 A="g2a2 g.a8&a.g8& g8gg8aa g.a8&a2 b-2>c2 <b-.>c8&c4.<b-8&
b-8b-b-8>cc <b-.>c8&c2<
1170 p(1)=A+"g2a2 g.a8&a.g8& g8gg8aa baga v11
1180 p(2)=STRING$(2,"l4b2b.b8 r6b12r6b12>c<b g2.r6b12& bag8a.")
1190 p(3)="l2v13o4 g1&gg a1&aa a1&aa g1&gg g1&gg a1r8a4.>c+ c1&c
<g g1r8g4.b
1200 p(4)="g1a>c+< g1aa g2.g4ae b4.a8&a4b4&b6a12&a2.
1210 p(5)="v1l1l ggg b4.b8&b2 g2r8g4. b1
1220 p(6)="l1v12gfedgfe2.r6b12&b4>c4c+4d4<
1230 p(7)="i3o3l12v13 d4&d6e-&e-4&e-6e-&e-4&e-6f&f2
1240 p(8)=p(0)+A+A
1250 p(9)="l12 g6gr2g6g r1 g6gr2g6g l6rdd+eff+ v11
1260 p(10)="v13 g.g.f.f.d+d+d4.> l48v15 p1gab>d2&d16
1270 FOR i=0 TO 4:PLAY p(i);:NEXT:PLAY "v12"+p(2);
1280 FOR i=5 TO 6:PLAY p(i);:NEXT
1290 FOR i=3 TO 4:PLAY p(i);:NEXT:PLAY "v12"+p(2);:PLAY p(5);
1300 FOR i=7 TO 9:PLAY p(i);:NEXT
1310 PLAY p(2)+p(5);:PLAY p(2)+p(10);
1320 FOR i=0 TO 20:p(i)="":NEXT
1330 PLAY ":";
1340 '
1350 p(0)="i5q8l4 v14k5o4=0 p3
1360 A="e2f+2 e.d8&d.e8& e8ee8f+f+ e.f+8&f+2 g2a2 g.a8&a.g8& g8g
g8aa g.a8&a2
1370 p(1)=A+"e2f+2 e.f+8&f+.e8& e8ee8f+f+ gf+ef+ v11
1380 p(2)=STRING$(2,"l4g2g.g8 r6g12r6g12gg e2.r6g12& gf+e8f+.")
1390 p(3)="l2v13o4 d1&dd e1&ee e1&ee d1&dc8<b4.> d1&dd e1r8e4.a
a1&ae d1r8d4.g
1400 p(4)="d1ea d1ee d.d4ec+ f+4.f+8&f+4f+4&f+6f+12&f+2.
1410 p(5)="l1 cde e4.e8&e2 e2r8e4. f+1
1420 p(6)="l1v12dcc<b>dcc2.r6g12&g4a4a+4b4
```


▶ NEW X68000にひとこと。EXPERT IIだろうが，SUPERだろうが目標はひとつ。日本一使いやすいパーソナルワークステーションになることだ。
段　宏太郎（19）X68000EXPERT　兵庫県

```
1430 p(7)="i3o2l12v13 b4&b6b-&b-4&b-6>c&c4&c6d-&d-2
1440 p(8)=p(0)+A+A
1450 p(9)="l12 e6er2e6e r1 e6er2e6e r12_6 l6rdd+eff+12 v11
1460 p(10)="v13 e-.e-.d-.d-.<bba4.>>l48rv13p1gab>d2&d16
1470 FOR i=0 TO 4:PLAY p(i);:NEXT:PLAY "v12"+p(2);
1480 FOR i=5 TO 6:PLAY p(i);:NEXT
1490 FOR i=3 TO 4:PLAY p(i);:NEXT:PLAY "v12"+p(2);:PLAY p(5);
1500 FOR i=7 TO 9:PLAY p(i);:NEXT
1510 PLAY p(2)+p(5);:PLAY p(2)+p(10);
1520 FOR i=0 TO 20:p(i)="":NEXT
1530 PLAY ":";
1540 RUN"0:ﾊﾟﾚｰﾄﾞ ｼｮｳｮ 3.mml
```

リスト4 パレードしようよ 3.mml

```
10 '    PROGRAM 3  (Ch8-Ch11)
20 DEFSTR a-z:DEFINT i:DIM p(30)
30 '--------------- make drums ------------
40 bd="i6c&<b&b-&a&a-&g&g-r>
50 t1="g&g-&f&e&e-r@1
60 t2="f&e&e-&d&d-r@1
70 t3="d&d-&c&<b&b->r@1
80 tm="f&e&e-&d&d-&c&<b&b->
90 t4="g&g-g&g-&f&e&e-&d
100 t5="f&ef&e&e-&d&d-&c
110 '------------- drum -----------
120 p(0)="i6q8l64@v127 o3=0
130 p(1)=bd+"r8i8"+t4+"r8 r4 i7e32e8.r32 r4i8"+t5+"r8 r6l96"+tm+
"d&d-&c&<b&b-&a&a-&g>r6 l64
140 p(2)="r4i8"+t4+"r8 r4 i7e32e8.r32 r4i8"+t5+"r8 r4 i7e4
150 p(3)="r4i8"+t4+"r8 r4 i7e32e8.r32 r4i8"+t5+"r8 r6l96"+tm+tm+
"r12"+bd+"l64
160 p(4)="r4i7e32e8.r32 r4i8"+t1+"r12"+t1+"r12"+STRING$(4,t1)+ST
RING$(4,t2)+t3+"r12l96"+bd+"l64
170 p(5)="r4 i8"+t5+"r8 r4 i7e32e8.r32 r4i8d&d- d&d-&c&<b&b-&a>r
8 r6l96"+STRING$(2,"g&g-&f&e&e-&d&d-&c")+"r12"+bd+"l64 r4i7e4r4i
8"+t5+"r8r4i7e4r6l96"+bd+" l64 i7e4
180 p(6)=bd+"r8i7e4"+bd+"r8i7e6l96"+bd+"r6"+bd+"l64i7e4"+bd+"r8i
7e4
190 p(7)=bd+"r8i7e4"+bd+"r8i7e6l96"+bd+"r6"+bd+"i7e4"+bd+"r12l64
i7e12e6e12
200 p(8)=bd+"r8i8"+tm+"r8"+bd+"r8i7e4
210 p(9)=bd+"r8i8"+tm+"r8l96"+bd+"r12l64i8"+t1+"i7e4
220 p(10)=bd+"r8i8"+tm+"r8l96"+bd+"r12l64i7e12e4
230 p(11)=bd+"r8i7e4"+bd+"r8i7e4
240 p(12)=bd+"r8i7l12eee eee eeel64
250 p(13)=bd+"r8i7e8l96i8"+STRING$(3,"g&g-&f&e")+bd+"r12l64"+t1+
t2+"r6
260 p(14)=bd+"r8r2. r2.r6l96"+bd+"l64"+bd+"r8r2. r1"+bd+"r8r2. r
2.r6l96"+bd+"l64"+bd+"r8r2i7e48e12.&e48l96"+bd+"r6"+bd+"l64i7e4r
6e12e6e12
270 p(15)="i8"+STRING$(6,t1)+STRING$(6,t2)
280 p(16)="l96"+bd+"r12i7e12e6"+bd+"i7r6e12e6"+bd+"r6l64i8"+t1+t
3+"r12l96"+bd+"r6i7e12e6"+bd+"l64
290 p(17)="r4i8"+t4+"r8r4 i7e32e8.r32 r4i8"+t5+"r8 r6l96"+tm+"d&
d-&c&<b&b-&a&a-&g>r12"+bd+"l64
300 p(18)="r4i8"+t4+"r8r4 i7e32e8.r32 r4i8"+t5+"r8 r6l96"+bd+"i7
e4l64
310 p(19)=p(3)+"r4i7e32e8.r32l96"+bd+"i8"+STRING$(3,t5+"r12")
320 p(20)="f&ef&e&e-&d&d-&cl64"+t1+t1+t1+t2+t2+t2+t2+t3+"r12l96"
+bd+"l64
330 p(21)=p(17)+p(18)
340 p(22)=p(3)+"r4i8"+tm+"r8r4i7e32e8.r32 r6e12e4e6e12e6l96"+bd+
"l64
350 p(23)="l96"+bd+"r12"+bd+"r6"+bd+"i7e4"+bd+"r12"+bd+"r6"+bd+"
r6"+bd+"i7e6"+bd+"l64"+bd+"r8l96
360 p(24)=bd+"r12"+bd+"r6"+bd+"i7e4e6e12 r6"+STRING$(5,bd+"r12")
+"l64
370 p(25)="l96r6"+bd+"i7l12e6e eeeeeel64
380 p(26)=bd+"r8i7e4"+bd+"r8i7e6l96"+bd+"i7l12r4e6e eeeel64i8"+t
1+t1
390 p(27)="l96"+bd+"r12l64"+t1+t1+"r12l96"+bd+"r4i7e4 r6"+bd+"r4
l64"+bd+"r8"+bd+"r8i7e4r2.
400 '
410 FOR i=0 TO 5:PLAY p(i);:NEXT
420 FOR i=1 TO 3:PLAY p(6);:NEXT:PLAY p(7);
430 FOR i=1 TO 7:PLAY p(8);:NEXT:PLAY p(9);
440 FOR i=1 TO 7:PLAY p(8);:NEXT:PLAY p(10);
450 FOR i=1 TO 7:PLAY p(11);:NEXT:PLAY p(12);
460 FOR i=1 TO 4:PLAY p(6);:NEXT
470 FOR i=1 TO 5:PLAY p(11);:NEXT
480 FOR i=13 TO 14:PLAY p(i);:NEXT
490 ' 2ﾊﾞﾝ
500 FOR i=1 TO 7:PLAY p(11);:NEXT:PLAY p(9);
510 FOR i=1 TO 7:PLAY p(11);:NEXT:PLAY p(10);
520 FOR i=1 TO 7:PLAY p(11);:NEXT:PLAY p(12);
530 FOR i=1 TO 4:PLAY p(6);:NEXT
540 FOR i=1 TO 5:PLAY p(11);:NEXT
550 FOR i=15 TO 24:PLAY p(i);:NEXT
560 '
570 FOR i=1 TO 4:PLAY p(6);:NEXT
580 FOR i=1 TO 5:PLAY p(11);:NEXT:PLAY p(25);
590 FOR i=1 TO 3:PLAY p(6);:NEXT
600 PLAY p(26);:PLAY p(27);
610 '
620 FOR i=0 TO 30:p(i)="":NEXT
630 PLAY ":";
640 '------------ chorus & chord -------
650 p(0)="q8l4v13k0 o4   y7,28  ^0s4,1,5,0=3
660 A="c2d2 c.d8&d.c8& c8cc8dd c.d8&d2 e-2f2 e-.f8&f.e-8& e-8e-e
-8ff e-.f8&f2
670 p(1)=A+"c2d2 c.d8&d.c8& c8cc8dd ddcd l12s4,1,3,0=3
680 p(2)=STRING$(2,"o5v15r4b6bb6br6b r6br6b>c4<b4 r4g6gg6gr6b& b
4a4g6ar4")
690 p(3)="s4,1,5,0v14l2o3 b1&bb >c+1&c+c+ c1&cc <b1&br b1&bb> c+
1r8c+4.e e1&ec <b1r8b4.>d
700 p(4)="<b1>c+e <b1>c+c+ <b2.b4>c+<a >v15c4.c8&c4c4&c6c12&c2.v
14
710 p(5)="s4,1,5,0 o3l1v12 ab>c c+4.c+8&c+2 c2r8c4. c1
720 p(6)="s4,1,5,0 o3l1v13 baggba g2.r6>d12&d4e4e+4g4
730 p(7)="l12v13 g4&g6g&g4&g6a-& a-4&a-6a-&a-2
740 p(8)=p(0)+A+A
750 p(9)="s4,1,5,0o4v13l12 c6cr2c6c r1 c6cr2c6c l6rdd+eff+  s4,1
,3,0l12
760 p(10)=LEFT$(p(5),LEN(p(5))-2)+"d1 s4,1,3,0l12
770 p(11)="s4,1,3,0o3l4v14 b-.>c8&c<a-& a-8b-.f+g+ f+1
780 FOR i=0 TO 4:PLAY p(i);:NEXT:PLAY "l12s4,1,3,0"+p(2);
790 FOR i=5 TO 6:PLAY p(i);:NEXT
800 FOR i=3 TO 4:PLAY p(i);:NEXT:PLAY "l12s4,1,3,0"+p(2);:PLAY p
(5);
810 FOR i=7 TO 9:PLAY p(i);:NEXT
820 PLAY p(2);:PLAY p(10);:PLAY p(2);·PLAY p(11);
830 FOR i=0 TO 20:p(i)="":NEXT
840 PLAY ":";
850 '
860 p(0)="q8l12o4k1v15 ^0s4,1,5,0=3
870 p(1)=STRING$(12,"r1")+"s4,1,3,0=3
880 p(2)=STRING$(2,"o5v15 r4g6gg6gr6g r6gr6ga4g4 r4e6ee6er6g& g4
f+4e6f+r4")
890 p(3)=STRING$(16,"r1")
900 p(4)="v13l2o3s4,1,2,0=3 b>dc+1 <b>dc+1 <b>dc+<a >d1&d1
910 p(5)="s4,1,5,0 o4l12v13 a4>c6g&g2 <b4>d6g&g2 c4e6g&g2 c+4e6g
&g2<<g1a1
920 p(6)="s4,1,4,0 o4l12v15 r1r1r1r1 r2.r6f& f6de6fr6de6f v14r2.
r6d&d4e4e+4g4
930 p(7)="l12v13 d4&d6e-&e-4&e-6e-& e-4&e-6f&f2
940 p(8)="s4,1,3,0o4l12v13"+STRING$(9,"r1")+"r6<b>c6ef+4g6a r1 b
6b-a6a-g6ef4 r2e-6de-6g f4e-6de-6gf4 r1 r6e-f6a>c4<r4
950 p(9)="s4,1,5,0o4v13l12 g6gr2g6g r1 g6gr2g6g r1 s4,1,3,0l12
960 p(10)=p(5)
970 FOR i=0 TO 4:PLAY p(i);:NEXT:PLAY "l12s4,1,3,0"+p(2);
980 FOR i=5 TO 6:PLAY p(i);:NEXT
990 FOR i=3 TO 4:PLAY p(i);:NEXT:PLAY "l12s4,1,3,0"+p(2);:PLAY p
(5);
1000 FOR i=7 TO 9:PLAY p(i);:NEXT
1010 PLAY p(2);:PLAY p(10);:PLAY p(2);:PLAY p(11);
1020 FOR i=0 TO 20:p(i)="":NEXT
1030 PLAY ":";
1040 '------------ hi-hat ----------
1050 FOR i=0 TO 20:p(i)="":NEXT
1060 hc="s4,1,15,0=3y6,4
1070 ho="s4,3,4,0=3y6,2
1080 cr="¯ls4,3,4,0=3y6,9 c4_1
1090 p(0)="q8l4o4v14k0^0s4,1,15,0=3y6,4
1100 p(1)=cr+hc+STRING$(6,"c")+"c6"+cr+hc+"r12cccccc"+cr+hc+"ccc
cccc6"+cr+hc
1110 p(2)="r12cccr2.r6"+cr+hc+"r12cccccccr6"+cr+hc+"r12cccr"+cr+"
rc"+hc
1120 p(3)=cr+hc+STRING$(15,"c")+cr+hc+STRING$(10,"c")+"c6"+cr+hc
+"r12ccr
1130 p(4)=hc+"cccc
1140 p(5)=hc+"crcr"+cr+hc+"rc"+ho+"c
1150 p(6)=hc+"c"+ho+"c"+hc+"cr
1160 p(7)=cr+hc+STRING$(15,"c")+cr+hc+"ccccccc"+cr+hc+"cc"+cr+hc
+"crrr
1170 p(8)=cr+cr+hc+STRING$(14,"c")+cr+hc+STRING$(15,"c")
1180 p(9)=cr+hc+"ccc cccc cccc ccc"+cr+hc+"cccr ccrr
1190 p(10)=STRING$(3,cr+ho+"v15crcrcrc")+cr+ho+"v15crr6"+cr+"r12
v15crrv14
1200 p(11)=cr+"r6c12&c6r6.c12& c6r6.c12&c6r6.c12&
1210 A="c"+hc+STRING$(13,"c")+"rr"+STRING$(7,"c")+"c6"+cr+"r12"
1220 p(12)=A+"rrrrrrr"+A+hc+"ccccrrr
1230 p(13)="¯ls4,3,4,0y6,9 c6c12&c_1"+ho+"c¯ls4,3,4,0y6,9c6c12_1
"+hc+"cccc
1240 p(14)=hc+"c6"+ho+"c12&ccr s4,3,4,0y6,9¯lr6c6c6c6c6_1
1250 p(15)=cr+cr+hc+"cr"+STRING$(26,"c")+"rr"+cr+"r6"+cr+"r12"+c
r+"r6"+cr+"r12"+cr+cr+cr
1260 p(16)="s4,1,5,0y6,7l32"+STRING$(24,"c")
1270 '
1280 FOR i=0 TO 3:PLAY p(i);:NEXT
1290 FOR i=1 TO 7:PLAY p(4);:NEXT:PLAY p(5);
1300 FOR i=1 TO 6:PLAY p(4);:NEXT:PLAY p(6);
1310 FOR i=7 TO 10:PLAY p(i);:NEXT
1320 '
1330 FOR i=1 TO 7:PLAY p(4);:NEXT:PLAY p(5);
1340 FOR i=1 TO 6:PLAY p(4);:NEXT:PLAY p(6);
1350 FOR i=7 TO 9:PLAY p(i);:NEXT
1360 FOR i=11 TO 14:PLAY p(i);:NEXT
1370 FOR i=8 TO 9:PLAY p(i);:NEXT
1380 PLAY p(15);:PLAY p(16);
1390 PLAY ""
1400 ' ===========================================================
===
1410 '          ﾊﾟﾚｰﾄﾞ ｼｮｳｮ                 PRINCESS PRINCESS
1420 '               programed by kazuhiko okada
1430 ' ===========================================================
===
```

▶ここのことを書くのにいつも本に書いてあることを参考にと読むと，Oh！Xを熟読したことになる。
関口　敬文（15）X68000EXPERT　兵庫県

★(で)のショートプロぱーてぃ

夜,見ないよーに

Komura Satoshi 古村　聡

今月の作品は，X1用穴掘りゲームと，画面が「ウネウネ」と変化するX68000用環境ソフト２本。うち１本はショートプロ初のC言語です。が，(で)氏曰く「リストも短くX-BASIC感覚で読める」のでわかりやすいと思いますよ。

illustration:T.Takahashi

(Q)完成したプログラムが作れないのですが……，それでも投稿していいですか？

(A)今月は質問ハガキがきたのでいきなりQ＆Aから始まってしまいました。たまにはいいですねぇ，こういうのも。ふぉふぉふぉっ。で，回答ですが。うーむ，ジュースが箱ごと送られてきたりするこのコーナーに送っていけないものなど存在するだろうか……？　よっぽど特殊なものでなければ，なにを送ってもへーキ，「どっからでもかかってきなさいっ！」(おお，なつかしい)。でも，やっぱり掲載率は完成したプログラムのほうが絶対高いですけど。なんてたってショートプロぱーてぃの主旨（そんなもの，あったのか？）は「みんなで楽しくぱぁーっとプログラムでも作ろうよ！」ですからね，ぱぁーっとノリで一本完成させてくださいよぉ，ね。でも，新コーナーの企画とか面白い話とかでもうれしいな，と思う私ですけど。

というわけでパズルゲーム

というわけで，ぱあーっといきます。今月の１作目は東京都の福田さんによるX1用パズルゲーム「DIGMAN」（リスト１）です。

DIGMAN　For X1

(CZ-8FB01)

東京都　福田強

下のような「休めっ！」の姿勢をしているような主人公を動かして水色の宝を取りつつ進んでください。

DIGMAN.BAS

全部取れれば面クリア。４，６キーで動かして１，３キーで穴が掘れます。青い壁の部分は掘れません。２，８のキーでハシゴの上下に動けますが，ハシゴの下になにもない場合主人公は落ちてしまいます。また，緑色の土の部分は１回通過すると消えてしまいます。そうそう，ギブアップはスペースキーです。さぁ，宝を掘りまくって

リスト1　DIGMAN.BAS

```
10 INIT:WIDTH 40:CLS0:DEFINT A-Z:CLICK OFF:DIM C$(5),M(12,8)
20 R$=CHR$(31,29,29,29):M$=" ● "+R$+"(■)"+R$+"／ ＼":FOR I=0 TO 5
30 READ A$,B$,C$:C$(I)=A$+R$+B$+R$+C$:NEXT:FOR I=1 TO 12
40 M(I,8)=1:NEXT:FOR I=0 TO 8:M(1,I)=1:M(12,I)=1:NEXT
50 '*****  TITTLE
60 LOCATE 16,11:PRINT "DIGMAN":LOCATE 18,13:PRINT "by T.F."
70 IF STRIG(0)=0 THEN 50 ELSE CLS
80 R=1:PLAY 200:SOUND 7,56:SOUND 12,5:SOUND 13,0:PLAY"V16O6R0"
90 '*****  SCREEN
100 RESTORE 480:FOR I=1 TO R:READ A$,B$,X,Y,T:NEXT:A$=A$+B$
110 FOR I=1 TO 7:FOR J=2 TO 11:M(J,I)=VAL(MID$(A$,I*10+J-11,1))
120 NEXT J,I:CLS:FOR I=0 TO 7:FOR J=1 TO 12:LOCATE J*3-1,I*3
130 COLOR M(J,I):PRINT C$(M(J,I)):NEXT J,I:COLOR 1
140 LINE(2,24)-(37,24),"■":COLOR 7:LOCATE 16,24
150 PRINT USING"ROUND:##";R;:LOCATE X*3-1,Y*3:PRINT M$
160 '*****  MAIN
170 IF STRIG(0) THEN 410
180 IF M(X,Y+1)=0 THEN XX=0:YY=1:GOTO 290
190 S=STICK(0):IF S<>1 AND S<>3 THEN 250
200 '---  DIG
210 XX=(S=1)-(S=3):IF M(X+XX,Y+1)<>2 THEN 170
220 M(X+XX,Y+1)=0:FOR I=3 TO 5:LOCATE (X+XX)*3-1,Y*3+I
230 PRINT "   ":PLAY"C":NEXT:GOTO 170
240 '---  MOVE
250 XX=(S=4)-(S=6):YY=(S=8 AND M(X,Y)=3)-(S=2)
260 IF XX+YY=0 THEN 170
270 A=M(X+XX,Y+YY):IF A=1 OR A=2 THEN 170
280 IF A>3 THEN M(X+XX,Y+YY)=0:IF A=5 THEN T=T-1:PLAY"CEG"
290 LOCATE X*3-1,Y*3:COLOR M(X,Y):PRINTC$(M(X,Y)):X=X+XX:Y=Y+YY
300 LOCATE X*3-1,Y*3:COLOR 7:PRINT M$:PAUSE 1:IF T>0 THEN 170
310 '*****  CLEAR
320 LOCATE 17,13:COLOR 7:PRINT "CLEAR!":PLAY"C2DECEDGR0"
330 R=R+1:IF R<10 THEN 100
340 '*****  END
350 CLS:MSG$="CONGRATULATIONS!":GOSUB 380
360 MSG$="ALL ROUND CLEAR!":GOSUB 380
370 MSG$="GOOD-BYE!!":GOSUB 380:CLS:END
380 '---  MESSAGE
390 LOCATE 20-LEN(MSG$)/2,12:PRINT MSG$:PAUSE 20
400 LOCATE 0,12:PRINT STRING$(40," "):RETURN
410 '*****  MISS
420 LOCATE 17,13:COLOR 7:PRINT "MISS!!":PLAY"G2FD-BR0":GOTO 100
430 '*****  CHARACTER DATA
440 DATA "   ","   ","   ",███,███,███,███,▒▒▒,███
450 DATA ├─┤,├─┤,├─┤,▒▒▒,▒▒▒,▒▒▒,┌─┐,┌◆┐,└─┘
460 '*****  FIELD DATA
470 '     ----------**********----------**********
480 DATA 3212201113315000515331040522133212222503
490 DATA 315020110333222511135300000000,10,5,8
500 DATA 5555355553335555555355335535355355533553
510 DATA 335535553535555355503555355335,11,6,46
520 DATA 2202202222225222252252232222250222532222
530 DATA 025312252252322222522252222225,6,0,11
540 DATA 2252202443442444444434524425223542545444 3
550 DATA 052435224404444234445444454444,2,0,10
560 DATA 2222522222531125213534211252433452112543
570 DATA 342521124353125211352322252232,2,0,12
580 DATA 3222112223322111222332251152203222222223
590 DATA 022211222332251152233511111153,6,0,6
600 DATA 2211521113522222222321522522230212222223
610 DATA 212252212302112221231522252223,2,0,7
620 DATA 2202211233522122022521225312122223522522
630 DATA 221323211220212322225222532225,7,0,8
640 DATA 2124445443252444444432224444443251144222 3
650 DATA 221524250344102420034420242223,2,0,5
```


▶科学の進歩と未来，とパソコンを関連づけ「これからパソコンはどうあるべきか？」ということを読者に意識的に考えさせる目安となるようなページがほしい。または，パソコン使用手段の概念を拡げることを意図した記事がほしい。

坂本　雅洋(19) X68000PRO　奈良県

君も今日から大金持ちだっ！

で，なんとこのプログラム，この短さで9面まであります。やりますねぇ。460行からが面データだそうです。

そ，それにしても……，ふにゃーっ。む，ムズイっ。というより私の頭がパズル向きにできてないのかなぁ。「古村君って本当にパズルダメなんだねぇー」とか言われてしまったし。ク，クヤシイ。結局，このゲームで3日ツブしてしまった。

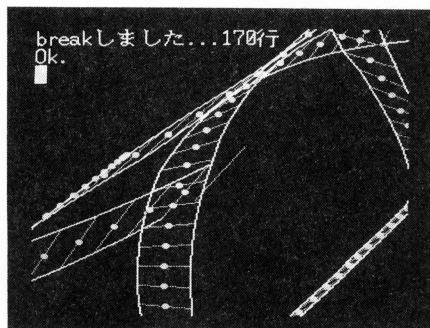
ところで，このゲーム，キャラクタのみでPCGもグラフィックも使ってないわけですが。いやー，キャラクタだけでもいろいろなキャラクターが作れてちゃんと見られる画面になるものなんですねぇ。なんか主人公がふんぞりかえっているように見えてこわいけど……。昔はこういった，キャラクタだけで作ったゲームが多かったんですが，私自身はこういうプログラムを作ったこともなければ遊ぶの初めてなのでおもわず「ははぁーっ」と拝んでしまいました。私もそうなんですが，最近のユーザーは初めて買ったマシンが640×400ドットのグラフィックだったり65536色だったりするので，こういうプログラムを見るとかえって新鮮さを感じるかもしれませんねー（え，私だけですか，感じるの!?）。

こりゃ，かわったデモだ！

今月の2，3作目は東京都（おおっ，今月は2人とも東京だ！）の太田さんの「空飛ぶDNAデモ」（リスト2）と「夜中にひとりで見てはいけないデモ」（リスト3）です。名前がなかったので，勝手に私がつけてしまいました。

空飛ぶDNAデモ　For　X68000
(X-BASIC)
夜中にひとりで見てはいけないデモ
For　X68000
(C Compiler PRO-68K)
東京都　太田敬三

2作ともデモプログラム（環境ソフトというのかな？）なんですが，「空飛ぶ……」のほうはX-BASIC，「夜中……」のほうは，なんとこのコーナー初のC言語での投稿です（おおっ！）。

で，デモの内容なんですが，「空飛ぶ……」のほうは画面にらせん階段のあまりらせんしてないヤツ（なんだそりゃ!?）を描いていくプログラムです。乱数を使って美しい図形を画面いっぱいに描いてくれます。夜中，電気を消して見てみるとなかなか芸術してます。

空飛ぶDNAデモ

夜中にひとりで見てはいけないデモ

反対に「夜中……」のほうは……。言えないっ，私の口からはとても言えないっ！いや，それ以上になんと表現してよいものかわからないっ！　そうあえて表現するなら……そう，悪魔の芸術とでも言えばいいのだろうか……（ますますワケがわからなくなりそうだな）。とりあえず自分で打ち込んで楽しんでくれたまえ。

で，C言語のプログラムの打ち込み方法を紹介するまえに，BASICしか知らない方にCコンパイラとはどういうものなのかをかるーく解説しときましょうね。

BASICの場合プログラムを動かすには，

1)　BASICを立ち上げる。
2)　行番号とプログラムを入力する。
3)　RUNする。
4)　BASICがプログラムを読んで動く。

と，すればよかったわけですけど，実は「行番号をつけながらプログラムを入力」というのはBASICのなかにエディタという部分が内蔵されているからできてたんですよね。が，残念なことにX68000のXCにはエディタが内蔵されていない（っていうか，普通はCってこういったものなんですね。最近はMS-DOSのTurboCとかエディタでRUNするものもあるけど）。そこでエディタでプログラムを入力してからCコンパイラでコンパイル，さらに作ったプログラムを動かすという，BASICなら自動でやってくれることをぜーんぶ手作業でやらなくちゃいけないわけです。

で，具体的な打ち込み方法は，

1)　ED GONBE2.Cとエディタを起動します。
2)　プログラムリストを打ち込みます。
3)　ESCキーを押してからEキーを押してエディタを終了します。

リスト2　空飛ぶDNAデモ(GONBE1.BAS)

```
 10 srand(333*val(mid$(time$,4,2))+777*val(right$(time$,2)))
 20 while 1
 30   screen 0,0,1,1:window(0,0,1023,1023):console ,,0
 40   home(0,384,384):box(0,0,1023,1023,14)
 50   a=512:x=a:y=a:bx=a:by=a:blx=a:bly=a:brx=a:bry=a
 60   dx=rand() and 1023:dy=rand() and 1023:mx=(rand() and 15)+15:my=mx*3/4
 70   repeat:c0=rand() and 15:c1=rand() and 15:c2=rand() and 15:until c0+c1+c2
 80   for z=0 to 500
 90     if rand()<3000 then dx=rand() and 1023:dy=rand() and 1023
100     vx=vx+sgn(dx-x)+(vx>mx)-(vx<-mx):x=x+vx
110     vy=vy+sgn(dy-y)+(vy>my)-(vy<-my):y=y+vy
120     a=y-by:lx=x+a:rx=x-a
130     a=x-bx:ly=y+a:ry=y-a
140     hx=x-128:hx=hx+1024*((hx>1023)-(hx<0))
150     hy=y-128:hy=hy+1024*((hy>1023)-(hy<0))
160     home(0,hx,hy)
170     line(lx,ly,rx,ry,c1):fill(x-1,y-2,x+1,y+2,c0):box(x-2,y-1,x+2,y+1,c0)
180     line(blx,bly,lx,ly,c2):line(brx,bry,rx,ry,c2)
190     blx=lx:bly=ly:brx=rx:bry=ry:bx=x:by=y
200   next
210   contrast(0):po():box(1,1,1022,1022,15):img_scrn(2,0,1):home(0,1,1)
220   contrast(15):po()
230   img_home(1,1,4,254,9000):po():img_home(255,1,8,510,9000):po()
240   img_home(255,511,6,254,9000):po():img_home(1,511,2,255,9000):po()
250   img_home(1,256,4,127,9000):po()
260   contrast(0):po()
270 endwhile
280 end
290 func po():for z=0 to 3000:next:endfunc
```

▶現在の私の推薦アニメは「トップをねらえ！」です。浦島効果，妙に生活感のある宇宙戦艦，リニア新幹線の中で流れる鉄道唱歌のメロディなど「設定おたく」にとっては，こたえられない作品なのです。　玉野　健一（19）PC-8801mk2MR　奈良県

4)　ディスク上にGONBE2.Cというプログラムがセーブされているので，コマンドラインから，

CC /O/Y/W GONBE2.C

と入力。するってーとCコンパイラがプログラムが動くようにコンパイルしてくれる。あえていうならば，4) と6) がBASICのRUNにあたるわけですね。厳密にはちょっと違うけど。

5)　入力ミスがあるとここでエラーが出ます。そうしたら1)に戻ってプログラムをじっと見て間違いを直して3) をやって……，とエラーが出なくなるまで繰り返します。

6)　そうして，GONBE2.Xがディスクにできたら，

GONBE2

と入力。……やった，プログラムが動いた！　V (ぶいっ)！

となるわけです。けっこう簡単でしょ。ま，Cは別売りだからみんなが持っているとは限らないのですが，XCはお買いドクですよぉ。BASICコンパイラ (略してBC) が付いてきますからねー。私もBCはスゴく重宝してます。なんてたってBASICでプログラムを作るとCやアセンブラのプログラムも一緒にできてきますもんねー。あぁ，それにしてもエディタ内蔵のCコンパイラ，どこかが出してくれるとうれしいんだけどなー。楽ですからね，あれ。CじゃなくてPAS

リスト3　夜中にひとりで見てはいけないデモ(GONBE2.C)

```
  1: void main()
  2: {
  3:         static char sp0[256] = {
  4:                 0,0, 0, 0, 0, 0,12,13,13,14, 0, 0, 0, 0,0,0,
  5:                 0,0, 0, 0,11,12,12,13,13,14,14,15, 0, 0,0,0,
  6:                 0,0, 0, 0,11,12,12,13,13,14,14,15, 0, 0,0,0,
  7:                 0,0, 0,11,11,11,12,13,13,14,15,15,15, 0,0,0,
  8:                 0,0, 0,10,11,11,12,13,13,14,15,15,14, 0,0,0,
  9:                 0,0,10,10,11,11,12,12,13,14,14,14,14,14,0,0,
 10:                 0,0, 9,10,10,11,11,12,13,13,14,14,14,13,0,0,
 11:                 0,0, 9,10,10,11,11,12,12,13,13,13,13,13,0,0,
 12:                 0,0, 9, 9,10,10,11,11,12,12,13,13,13,13,0,0,
 13:                 0,0, 9, 9,10,10,11,11,11,12,12,12,13,13,0,0,
 14:                 0,0, 8, 9, 9,10,10,11,11,11,12,12,12,12,0,0,
 15:                 0,0, 0, 8, 9,10,10,10,11,11,11,12,12, 0,0,0,
 16:                 0,0, 0, 7, 9, 9,10,10,10,11,11,11,11, 0,0,0,
 17:                 0,0, 0, 0, 7, 9, 9,10,10,10,11,11, 0, 0,0,0,
 18:                 0,0, 0, 0, 7, 8, 9, 9,10,10,10,11, 0, 0,0,0,
 19:                 0,0, 0, 0, 0, 0, 8, 9, 9,10       };
 20:         static char sp1[256] = {
 21:                 0,0, 0, 0, 0, 0,12,13,13,14, 0, 0, 0, 0,0,0,
 22:                 0,0, 0, 0,11,12,12,13,13,14,14,15, 0, 0,0,0,
 23:                 0,0, 0, 0,11,12,12,13,13,14,14,15, 0, 0,0,0,
 24:                 0,0, 0,11,11,11,12,13,13,14,15,15,15, 0,0,0,
 25:                 0,0, 0,10,11,11,12,13,13,14,15,15,14, 0,0,0,
 26:                 0,0,10,10,10,11,11,12,14,15,15,14,14,14,0,0,
 27:                 0,0, 9,10,10,10,11,12,14,15,14,14,14,13,0,0,
 28:                 0,0, 9, 9, 9, 9,10,11,15,14,13,13,13,13,0,0,
 29:                 0,0, 9, 9, 9, 9, 8, 7,11,12,13,13,13,13,0,0,
 30:                 0,0, 9, 8, 8, 8, 7, 8,10,11,12,12,12,13,0,0,
 31:                 0,0, 8, 8, 8, 7, 7, 8,10,11,11,12,12,12,0,0,
 32:                 0,0, 0, 8, 7, 7, 8, 9, 9,10,11,11,12, 0,0,0,
 33:                 0,0, 0, 7, 7, 7, 8, 9, 9,10,11,11,11, 0,0,0,
 34:                 0,0, 0, 0, 7, 8, 8, 9, 9,10,10,11, 0, 0,0,0,
 35:                 0,0, 0, 0, 7, 8, 8, 9, 9,10,10,11, 0, 0,0,0,
 36:                 0,0, 0, 0, 0, 0, 8, 9, 9,10       };
 37:
 38:         register  int no,a;
 39:         int b,co,pl,sx,sy,pd,
 40:             ox[32],x[32],dx[32],vx[32],dvx[32][4],
 41:             oy[32],y[32],dy[32],vy[32],dvy[32][4];
 42:
 43:         CRTMOD( 2); B_CUROFF();
 44:
 45:         sp_init(); sp_disp( 1); sp_on( 0, 127);
 46:         for( a=0; a!=3; a++) sp_def( a, sp1, 1);
 47:         sp_def( 3, sp0, 1);
 48:
 49:         for( a=7; a!=16; a++)
 50:             TPALET2( a, hsv( 6, 52-a*3, a*3-14) );
 51:         for( b=1; b!=16; b++)
 52:             for( a=6; a!=17; a++)
 53:                 sp_color( a,hsv( b*12+6,52-a*3,a*3-14), b);
 54:
 55:         bg_set( 0, 0, 1); bg_fill( 0, 255);
 56:         bg_scroll( 0, 488, 490);
 57:         for( a=0; a!=32; a++)
 58:         {
 59:             b = a << 1; if( a>15 ) b -= 31;
 60:             ox[b] = 48 * ( a / 7 ) + 40;
 61:             oy[b] = 64 * ( a % 7 ) + 38;
 62:             if( oy[b] > 255 )  ox[b] += 24, oy[b] -= 224;
 63:             sx = ox[b] - 40 >> 3;
 64:             sy = oy[b] - 38 >> 3;
 65:             pd = b >> 1 << 8;
 66:             bg_put( 0, sx  , sy  , pd  );
 67:             bg_put( 0, sx  , sy+1, pd+1);
 68:             bg_put( 0, sx+1, sy  , pd+2);
 69:             bg_put( 0, sx+1, sy+1, pd+3);
 70:         }
 71:
 72:         for( a=0; a!=32; a++)
 73:         {
 74:             x[a] = y[a] = dx[a] = dy[a] = 8;
 75:             vx[a] = vy[a] = 0;
 76:         }
 77:
 78:         while( ! BITSNS(12) )
 79:         for( co=0; co!=4; co++)
 80:         {
 81:             for( no=0; no!=32; no++)
 82:             {
 83:                 if( rand()<500 ) dx[no]
 84:                         = rand() & 15, dy[no]
 85:                         = rand() & 15;
 86:                 dvx[no][co] = vx[no]
 87:                         += ( x[no] < dx[no] & vx[no] != 2)
 88:                         - ( x[no] > dx[no] & vx[no] !=-2 );
 89:                 x[no] += ( vx[no] > 0 ) - ( vx[no] < 0 );
 90:                 dvy[no][co] = vy[no]
 91:                         += ( y[no] < dy[no] & vy[no] != 3 )
 92:                         - ( y[no] > dy[no] & vy[no] !=-3 );
 93:                 y[no] += ( vy[no] > 0 ) - ( vy[no] < 0 );
 94:                 a = co;
 95:                 pl = ( no << 2 ) + 3;
 96:                 sx = ox[no];
 97:                 sy = oy[no];
 98:                 pd =  no >> 1 << 8;
 99:                 for( b=0; b!=4; b++, pl--)
100:                 {
101:                     a ++; if( a==4 ) a=0;
102:                     sx += dvx[no][a];
103:                     sy += dvy[no][a];
104:                     sp_set( pl, sx, sy, pd+b, 3);
105:                 }
106:             }
107:         }
108:         CRTMOD(16);
109: }
```

▶v_edit.basというBASICプログラムを作り，Cコンパイラでコンパイルしたら，「.c」「.s」「.o」ができた。そこで最近覚えたワイルドカードで消そうとdel v_edit.＊としたら，「.x」はおろか「.bas」まで消えてしまった。

川脇　泰宏 (16) X68000EXPERT　奈良県

CALだったりするとさらにラッキーだな(実は私はTurbo Pascal派なのだ)。でもTurbo Cみたいに標準のBASICやアセンブラのプログラムとリンクできなかったりすると困るし……。あぁ，マニアな会話。

で，話は戻ってこのデモなんですが。「空飛ぶ……」のほうはともかく「夜中……」はウゲゲであったりします（スタッフの一部から「きっ，気持ちいい！」という声もあったけど，ホントなに考えてんだか)。ぜひ自分の目で確かめてほしいですね，はい。プログラムの大きさもちょうどいいと思いますし。

ところで，実は，情けない話なんですが，私はCについてはほとんど初心者だったりするのです。はっはっはっ（笑ってる場合かっ！)。でも，私にも今回のCプログラムはぜーんぶ読めました。だってX-BASICとほとんど同じ感覚で読めるんだもんなー(このプログラムがポインタとかを使ってないせいもある)。X-BASICって偉大だなー，などと妙なところで感心してしまいました。感心してないで私もしっかり精進しなきゃだな，うん。

なにはともあれ，2本とも発想よし，短さよし，これが本当にショートかーっ!? というほどで，太田君にはもう参りましたっ！ 太田君にはこのままデモプログラムの道を極めていただきたいと思います。次の作品も期待してますからねーっ（さぁ，プレッシャーが)！

そんなところで今月も終わり。よい新学期を。また来月っ！

(で)のぱーてぃハンズ――――(その2)

諸君のみなさま，お待たせしましたぁ，ハンズのコーナーです。どうでもいいことなんですが，なーんかほかの記事とノリが違うこのコーナー。実は私は江戸むらさきのめんたいこ入りが好きで……。そのノリじゃないっつの！

なーんてつまらん冗談さえ許されるこのコーナー。でも，この気楽な姿勢こそがプログラムを組むのに必要なノリなんじゃないかと力説したい（そういうことだからろくなプログラムが組めないんだな，私は)。

さて，今月は先月作ったキャラクターをジョイスティックで動かしてしまおうという予定だったのですが……。しまったぁっ，その前に先月分について謝ってしまわなければならないのだった。

えー，今月号の投稿作品をみて思ったのですが。先月の私のプログラムのスプライトのパレット定義の部分に，DEFSPTOOL（買うとついてくるあれね）というスプライト定義ツールが作ってくれるパレット定義プログラムをそのまま行番号を変えて載せてしまったのです。が，考えてみたらあれで使っているパレットの0から7番は，黒・青・赤・マゼンタ・緑・水色・黄・白で，X1やなんかのカラーコードと同じくRGBが1，2，4になっている。だからそこの部分をうまくFOR文を使えばかなり短くなったはずなのですね。よって，もっとプログラムを短くしたいなーという方はそのようにしていただけると幸いです。ま，多少大きくなっても動くんだからべつにいいんですけど。

で，さらに先月，「自分好みにパターン変えてね」などと申しましたが，もしかしたら説明が足りなくて「で，できないっ！」などという人もいたかもしれないんですよね。ぜんぜん説明しなかったもんなー，はっはっはっ（笑いごとですむかっ！)。

そこで，パターンを変えたくても変えられなかった人のために，DEFSPTOOLでスプライトの絵を描いてからBASICのプログラムにするまでをちょっと解説してしまいましょう。アフターサービスも万全なのがこのコーナーの自慢なのです（ふぉふぉふぉっ，自画自賛してしまった)。あ，それから，このコーナーへの質問ハガキも待ってます。

はい，それではとりあえずBASICに入って先月号のリストの130行と1240〜2960行に，

```
140 END
```

を付け加えて打ち込んでSAVEしてください。そのあとRUNします。それから，

```
RUN "¥ETC¥DEFSPTOOL"
```

でDEFSPTOOLを立ち上げます。そうそう初代X68000の人は¥福袋で，ACEの人は¥ACEです。立ち上がりましたか？

さて，それでは右上の画面に絵を描いてパレットの色を変えてください（右下のパレットは今回は関係ないです。っていうより面倒だから私は右下のパレットは使ったことがないのだ。それはともかく，このあたりはだいたいわかるんじゃないかなー。最初に出てくる説明をしっかり読めば簡単ですから)。

PUTを2回クリックして絵を左側にあった元の絵と変えられました？ 番号は&H20から&H27になってます？ じゃ，あとはSAVEを2回クリックしてスプライトの絵のパターンとパレットの色の両方をセーブします。まずはパターンから。こっちはファイル名とパターンの番号を入力するんです。ま，ファイル名は自分さえ覚えられれば適当でいいんですが（たとえばXEVI.PATとか)，問題はパターン番号。これ実はあの左上の一覧表の上と左についてる数字で番号を入れようとするとパターン番号に&Hをつけてやらなくちゃいけないのでご注意を。

パレットのほうはファイル名（ここではXEVI.PLTとする)を入れるだけですね。で，パターンとパレットの定義ができたのですがこいつがまずいことにせっかく作ってくれた2つのプログラムの行番号が重なってしまっているのですね。で，しかたがないのでどちらか一方の行番号を変えます。このあたりは人によってやり方が違うかもしれませんが，私はED.Xを使います。まず，さっきのDEFSPTOOLでセーブし終わったらブレイクキーを押して，

```
WIDTH 96
```

と打ちます。

で，画面が変わったらパレットのロード。

```
LOAD "XEVI.PLT"
```

さて，いまロードしたプログラムには関係ないパレットの定義も入ってますので，

```
sp_color (4,65535,2)
                ↑
```

(ここの部分が1以外のものが関係ない)

この部分を消します。たぶん60170〜62400行がそれのはずです。

```
DEL 60170-62400
```

はい，これで消えました。このあと以下のように行番号を揃えます。

```
RENUM
```

そうして，以下のようにセーブしてから，

```
SAVE "XEVI.PLT"
```

SYSTEMと打ってBASICから抜けます。

さて，今度はEDの出番。ED XEVI.BAS（このファイル名も好きなものにしていいですよ）としてEDを立ち上げます。で，このEDにさっきの2つのファイルをロードします。ESCキーを押してからYを押してファイル名を入れるとロードしてくれますから，

```
[ESC] Y XEVI.PAT
[ESC] Y XEVI.PLT
```

とします。で，

```
10 SP_PALLET ():END
```

と，

```
10 SP_PATTERN ():END
```

という行がありますからこれを消します（消すのはDELでもBSキーでも使っていいけど，関係ない行まで消さないようにね)。

それが終わったら，

```
[ESC] E
```

でED.Xを終わります。うーむ，疲れる。でも，もうちょい！ で，もういちどBASIC.Xに入ってED.Xで作ったファイルをロード（LOAD "XEVI.BAS"ね）します。プログラムのパレットの定義は1240行からだから，

```
RENUM 1240
```

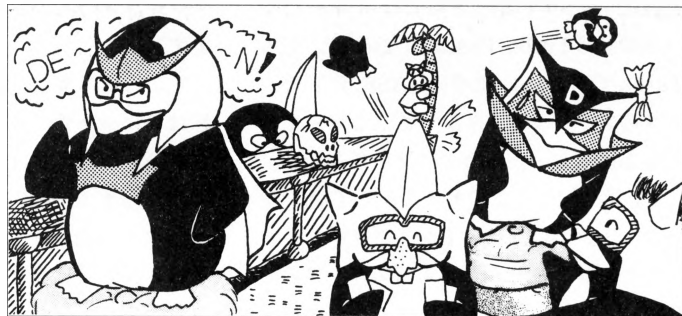
とすると，おお，先月号みたいなプログラムができてるじゃあないですか。あとは先月号の残りの部分を打ち込んでおーわりっと！

あー，お疲れさまでしたっと。こんなふうにするわけなんですね。おわかりいただけました？ 実は，私もまるっきり同じやり方であれを作っていたのです。はっはっはっ……（笑ってごまかす)。

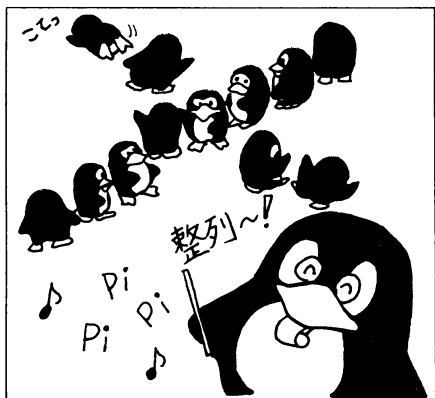
あ，しまった，今月はジョイスティックでスプライトを動かす予定だったのに，調子にのって書いてしまったらもうページがないじゃないかっ！ うーっ。来月こそは絶対にジョイスティック関係に進もう。進みたいなぁ，進むといいなぁ，進むかなぁ（思わず弱気になる私)。では，来月までおやすみなさい，ぐう。

▶X68000の新型が登場すると，旧型が安くなるのではないか？ 諦めていたX68000への野望が復活しそうである。　口井　雅弘（17）X1turboZ　和歌山県

マシン語カクテル in Z80's Bar
第11回――ライン文だべっちゃ!――

シナリオ：古村聡
特別監修：金子俊一　浦川博之
イラスト：山田純二

先月，ようこさんから「直線描画の話もお願いね」と言われた光君。その言葉をマに受けてラインルーチンを作ってきたのはいいけれど，ようこさんにプレゼントすることはできるのでしょうか？　努力の結果が夢で終わらなければいいのですが……。

♪カラン，コローン（ドアが開く音）
（オープニングの歌）
MOVをLDにかきかえて，
PUSHでレジスタ保存して，
CALLで呼べばー現れる。
よぉーびさきー，それはスタックにー，
あーいおー，それはBCでー，
4メガのはやいスピードー，
セグメントよりあついこーころー，
ぱっ，ぱっ，ぎゃくてんー，
ここからー16ビットはとおさなーい，
ぱっ，ぱっ，ぎゃくてんー，
これからー8ビットのはなみちー，
ぱっ，ぱっ，ぱっ，ぱっ，
ぱっ，ぱっ，ぱっ，ぱっ，
ぱっ！
ぜっぱちまーん！

純二（以下純）：さて，今回の指令だが。
ようこ（以下Yo）：な，なんで私がこんな格好しなくちゃならないのよぉーっ！
源光（以下光）：なんでって，今回はタ◯ムボ◯ンシリーズのパロディだから。全国の女子高生のみなさーん……なんて。
Yo：私，悪役なんてやだー！　ヒロインがいいー！
善司（以下善）：まぁまぁドロンジョ様。
Yo：だれがドロンジョよぉっ！

純：ゼッパチマンたちはX1時代のZ80's Bar地方へ向かったらしいのだ……。
Yo：だいたいなんなのよ，このセンスの悪い格好はー。
善：昔からタ◯ムボ◯ンの悪役はこのカッコに決まってまんがな，ドロンジョ様。
Yo：ドロンジョじゃないっていってるでしょ！
光：全国の女子高生のみなさーん。
純：今度こそゼッパチマンを倒し，その筋キーホルダーを手に入れるのだ……。ん，聞いてんのかおまえらはーっ！
Yo，善，光：アラホラサッサッ！

今月のプログラム

光：これが今月のびっくりどっきりメカでごじゃいまする。
Yo：あら，なんだいこれは。
光：X1用4連結アルゴリズムのラインルーチンです。BASICから呼び出してテキスト画面上にキャラクタでラインを引くんです。
Yo：あーら，やるじゃないかボヤッ◯ーッ！
光：ドロ…じゃない，ようこさんのために徹夜して作ってきたんですよ，はっはっは。
？：Oh！ワタシニモツクッテクレマスカァ？
光：（ドキッ！）も，もちろん，メアリーさんのためならたとえ火の中，水の中……，あれ，メアリーさんは？
善：Oh！ウレシイデース。うっわー，だーまされよったー。
光：きっ，貴様ーっ！　ボコッ！
善：ぽっくん。
Yo：いいから早くおしっ！
光：その口調もうやめましょうよ。それで，ラインっていうのは点（X1，Y1）から（X2，Y2）まで直線を引くんだけど，

$P=X2-X1$
$Q=Y2-Y1$

とおくと直線の方程式は，

$E=Q/P\ (X-X1)+Y1$

と書けるよね。で，ここから先月のサークルと同じように式変形していくんだ。
Yo：また数学なのねー，ゔえー。
光：でも，先月と基本的な考え方はほとんど変わらないからだいじょうぶ。また誤差がなるべく小さくなるようなドットを探して打っていくようにするんだ。スクリーン上の座標との誤差Eを表す式は，

$E=Q/P\ (X-X1)-P\ (Y-Y1)$

となるんだ(ただし$X1 \leqq X \leqq X2$)。でも，このとおりにやるのはめんどくさいから，また式変形で簡単にしてやるんだ。まずF=PEとすると，

$F=PE=Q\ (X-X1)-P\ (Y-Y1)$

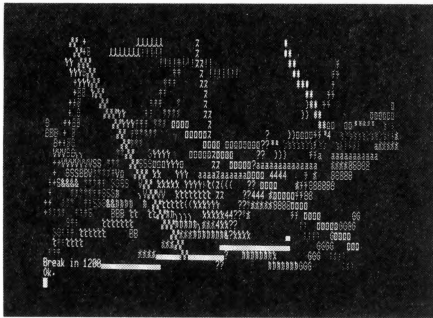
ドットが（X，Y）（誤差をFとする）から（X+1，Y）と（X，Y+1）に移動した場合の新たな誤差の式F'，F"を求めてみると，

1)　（X+1，Y）に移動した場合，
$F'=Q\,(X+1-X1)-(Y-Y1)=F+Q$
2)　（X，Y+1）に移動した場合，
$F''=Q(X-X1)-P(Y+1-Y1)=F-P$

となる。このとき実際にドットを打つときにはF'，F"のどっちか小さくなるほうを選ぶのでF'，F"の大小を比較するために，

$G=2F+2Q-P$

という式を使うんだ。
Yo：あら，この式は？
光：これはねー。実はこれを説明すると長くなるからねー。これはこういうものだと思っといて。
マスター（以下M）：しょうがない，マスターズメモに書いておくべぇー。
Yo：え，あ，あら!?　ね，光君，いまマスターいなかった!?

▶そうだったのか定期購読を申し込むよりも生協で買ったほうがだんぜんオトクなんだ。いいことを教えてもらった。へへへ。　森下　寛和（18）X68000，MZ-1500　鳥取県

光：たぶん幻覚だよ。あの人最近出番少ないから。で，話を続けると（X，Y）のときにFを比較する式Gが上の1）2)のときそれぞれG'，G"に変化する。G'，G"は，

$$
\begin{aligned}
1)\quad G' &= 2F'+2Q-P \\
&= 2\,(F+Q)+2Q-P \\
&= 2F+2Q-P+2Q \\
&= 2G+2Q \\
2)\quad G'' &= 2F''+2Q-P \\
&= 2\,(F-P)+2Q-P \\
&= 2F+2Q-P-2P \\
&= 2G-2P
\end{aligned}
$$

と書ける。2P, 2QはそれぞれP, Qを求めたときにそれぞれP+P，Q+Qをワークエリア上に取っておけば，めでたく直線を引くことができるわけ。

BASICで書いてみる

Yo：あ，じゃ，これを使ってまたBASICのプログラムにするのね。

光：そうだね。実際にラインを引くにはリスト1のようにするんだ。ちょっとif～thenのあたりがややこしいけどX1BASICで書いたのでわかるよね。で，これをマシン語化したのがリスト３（リスト２はそれを使用したBASICのサンプルプログラム）。はい，ようこさんにあげよう。

Yo：ありがとう，光君……。

光：ようこちゃん……（ししし，やった。もうここまでくれば，あれもできるこれもできる……ごっくん）。

善：あのー，よだれを垂らしてお取り込み中申し訳ないんだけど……。

光：（きりっ！）申し訳ないと思うんなら，ひっこまんかい！

善：もう，ゼッパチマンが登場してるんですけど……。

光：なにっ！

で：ゼッパチマン１号参上！

メアリー：オナジクゥ２ゴウサンジョウ！

で：Z80のある限り，この世に悪は栄えない！

光：こしゃくな。このボタンを押せば……。

リスト1　ラインルーチン(X1BASIC)

```
 10 REM LINE4(X1,Y1,X2,Y2,C)
 20 SCREEN 0,0,0
 30 INPUT"X1,Y1  :";X1,Y1
 40 INPUT"X2,Y2,C:";X2,Y2,C
 50 P=X2-X1
 60 P2=P*2
 70 X=X1
 80 Y=Y1
 90 PSET(X1,Y1,C)
100 IF Y1<Y2 THEN "\" ELSE "/"
130 LABEL"\"
140 Q=Y2-Y1:Q2=Q*2:G=Q2-P
150 FOR I=1 TO P+Q
160 IF G<0 THEN X=X+1:G=G+Q2
           ELSE Y=Y+1:G=G-P2
170 PSET(X,Y,C)
180 NEXT
190 END
200 LABEL"/"
210 Q=Y1-Y2:Q2=Q*2:G=P2-Q
220 FOR I=1 TO P+Q
230 IF G>0 THEN X=X+1:G=G-Q2
           ELSE Y=Y-1:G=G+P2
240 PSET(X,Y,C)
250 NEXT
260 END
```

ポチッ！　どっかーん！　ひゅるるるる。

Yo：ゲホッ，……なんでー，なんでこっちが自爆すんのよぉーっ。

光：いやー，ははは。なんかプログラムの組み方間違えて暴走しちゃったみたい……。

Yo：もう，ばかっ！

純：おーしおーきだべぇー。

で＆メアリー：しゅびびん，しゅびびん，しゅびびんびーん！

＊

で：……という夢を昨日，見てしまったりしたのだな私は。

善：なんだー，夢オチかー。

Yo：勝手に人を夢に登場させないでよねー。だいいち光君が女の子の気をひくため

リスト2　ラインルーチンデモプログラム

```
1000 '******************
1010 '*LINE   Demo     *
1020 '*    By Dec      *
1030 '*    1990,3/06   *
1040 '******************
1050 WIDTH 80
1060 '*var xr,yr,r,c : integer;
1070 '*    ch$               : char    ;
1080 '*begin {main}
1090  X1 = INT(RND(0)*80)
1100  Y1 = INT(RND(0)*25)
1105  X2 = INT(RND(0)*80)
1107  Y2 = INT(RND(0)*25)
1120  C  = INT(RND(0)*8  )
1130  CH = INT(RND(0)*256)
1135 COLOR C:GOSUB1150:COLOR7
1140  GOTO 1090
1150 '*end;  {main}
1160 '*procedure Line
1170 '*var x,y : integer;
1180 '
1190 '*begin {Line  }
1191 IF X1>X2 THEN SW=X1:X1=X2:X2=SW
1200  POKE &HE003,X1
1210  POKE &HE005,Y1
1211  POKE &HE007,X2
1212  POKE &HE009,Y2
1230  POKE &HE00B,CH
1240  CALL &HE000
1250  RETURN
1260 '*end   {Line  }
```

にプログラム作ってくるなんてマヌケなことするわけないじゃない，ねぇ，光君。

光：え，あ，もちろんですよ，きまってるじゃないですか，はははははは。（ガタン，とイスを動かす）ちょっと今日はこれで。

Yo：あら，帰っちゃうの。またねー。

M：ありがとうございました……（今月も出番，これだけか……）。

♪カラーン，コローン（ドアの閉じる音）

10分後，砂浜で夕陽に向かってようこさんのために作ってきたラインルーチンの入ったディスクを投げる光の姿があった。

光：ラインルーチンのばかやろーっ！

MASTER'S MEMO

今月のプログラムは４連結だがこれも同じような式変形で８連結のプログラムもできる。ただし，直線の場合も円と同じくその直線がY＝Xより上にあるか下にあるかで場合分けしなければならない。したがって４連結では２つでよかった場合分け（４連結は右上がりか右下がりかだけの場合しかない）が８連結では４つ必要である。これからもわかるように８連結では４連結のプログラムに比べてサイズが２倍程度になるはずである。ちなみにX1BASICのラインルーチンは８連結型のアルゴリズムで書かれているようである。また４連結型の例としてはFM-7のBASICのラインルーチンがある。

ところで本文中に出てきた，

$$G=2F+2Q-P$$

という式は８連結のアルゴリズムを作るときに導き出された式である。本文中の2)を（X+1，Y+1）に進むように変更すると，

$$
\begin{aligned}
F'' &= Q(X+1-X1)-P(Y+1-Y1) \\
&= F+Q-P
\end{aligned}
$$

となる。$|F'|<|F''|$であるかどうかを判定するのに，

$$
\begin{aligned}
F'^2-F''^2 &= (F'+F'')(F'-F'') \\
&= 2F+2Q-P \\
&= G
\end{aligned}
$$

と式変形してこれの正負を判定したうえで，次にドットを打つとき1)，2)どちらの動作で打つのか決定するのである。４連結のアルゴリズムも８連結では斜めに動くところを縦，横と動くだけだからこの式が使えるわけである。

最後に，先月号の式変形に一部まちがいがありました。P146右側の，

$$e=e'+X*2+1$$

は正しくは，

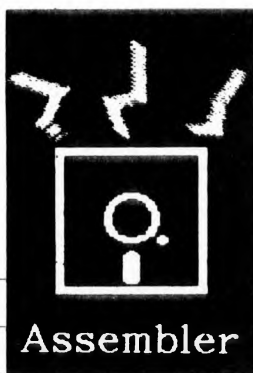
$$e=e'-X*2+1$$

です。大変申し訳ありませんでした。

▶最近，友人がX68000を買いました。今まで，家庭用テレビでパソコンをしていたのでとても喜んでいます。
久野　大祐（18）PC-8801FH　鳥取県

リスト3 ラインルーチン(マシン語)

```
 ORG $E000
E000 C3 73 E0     2 MAIN     JP       LMAIN
E003              3 ;WORKAREAS
E003 00 00        4 WX1      DW       0
E005 00 00        5 WY1      DW       0
E007 00 00        6 WX2      DW       0
E009 00 00        7 WY2      DW       0
E00B 00 00        8 WC       DW       0
E00D 00 00        9 WP       DW       0
E00F 00 00       10 WQ       DW       0
E011 00 00       11 WP2      DW       0
E013 00 00       12 WQ2      DW       0
E015 00 00       13 WX       DW       0
E017 00 00       14 WY       DW       0
E019 00 00       15 WG       DW       0
E01B             16 ;LOCATE & PRINT ROUTINE
E01B 14 00       17 WLX      DW       20
E01D 0A 00       18 WLY      DW       10
E01F 3A 1B E0    19 XYPRINT  LD       A,(WLX)
E022 FE 00       20          CP       0
E024 F8          21          RET      M
E025 FE 50       22          CP       80
E027 D0          23          RET      NC
E028 3A 1D E0    24          LD       A,(WLY)
E02B FE 00       25          CP       0
E02D F8          26          RET      M
E02E FE 18       27          CP       24
E030 D0          28          RET      NC
E031             29
E031 3A 1B E0    30          LD       A,(WLX)
E034 32 0E 00    31          LD       ($000E),A
E037 3A 1D E0    32          LD       A,(WLY)
E03A 32 0F 00    33          LD       ($000F),A
E03D 3A 0B E0    34          LD       A,(WC)
E040 CD C8 04    35          CALL     $04C8
E043 C9          36          RET
E044             37
E044             38
E044             39 ;LINE INIT
E044             40 ;P=X2-X1
E044 2A 07 E0    41 LINIT    LD       HL,(WX2)
E047 ED 5B 03    42          LD       DE,(WX1)
E04A E0
E04B B7          43          OR       A
E04C ED 52       44          SBC      HL,DE
E04E 22 0D E0    45          LD       (WP),HL
E051             46 ;P2=P*2
E051 29          47          ADD      HL,HL
E052 22 11 E0    48          LD       (WP2),HL
E055             49 ;X=X1
E055 2A 03 E0    50          LD       HL,(WX1)
E058 22 15 E0    51          LD       (WX),HL
E05B             52 ;Y=Y1
E05B 2A 05 E0    53          LD       HL,(WY1)
E05E 22 17 E0    54          LD       (WY),HL
E061             55 ;PSET(X,Y,C)
E061             56
E061 2A 15 E0    57 PSET     LD       HL,(WX)
E064 22 1B E0    58          LD       (WLX),HL
E067 2A 17 E0    59          LD       HL,(WY)
E06A 22 1D E0    60          LD       (WLY),HL
E06D CD 1F E0    61          CALL     XYPRINT
E070 C9          62          RET
E071             63
E071             64 ;LINE MAIN ROUTINE
E071 00 00       65 WI       DW       0
E073 CD 44 E0    66 LMAIN    CALL     LINIT
E076             67 ;IF Y1<Y2THEN
E076 2A 05 E0    68 IF1      LD       HL,(WY1)
E079 ED 5B 09    69          LD       DE,(WY2)
E07C E0
E07D B7          70          OR       A
E07E ED 52       71          SBC      HL,DE
E080 D2 E9 E0    72          JP       NC,ELSE1
E083             73 ;   (*
E083             74 ;      Q=Y2-Y1
E083 2A 09 E0    75 THEN1    LD       HL,(WY2)
E086 ED 5B 05    76          LD       DE,(WY1)
E089 E0
E08A B7          77          OR       A
E08B ED 52       78          SBC      HL,DE
E08D 22 0F E0    79          LD       (WQ),HL
E090             80 ;      Q2=Q*2
E090 29          81          ADD      HL,HL
E091 22 13 E0    82          LD       (WQ2),HL
E094             83 ;      G=Q2-P
E094 ED 5B 0D    84          LD       DE,(WP)
E097 E0
E098 B7          85          OR       A
E099 ED 52       86          SBC      HL,DE
E09B 22 19 E0    87          LD       (WG),HL
E09E             88 ;      FOR I=1 TO P+Q
E09E 21 00 00    89          LD       HL,0
E0A1 22 71 E0    90          LD       (WI),HL
E0A4             91
E0A4 21 71 E0    92 FOR1     LD       HL,WI
E0A7 34          93          INC      (HL)
E0A8             94
E0A8 2A 0D E0    95          LD       HL,(WP)
E0AB ED 5B 0F    96          LD       DE,(WQ)
E0AE E0
E0AF 19          97          ADD      HL,DE
E0B0             98 ;        LD       DE,HL
E0B0 ED 5B 71    99          LD       DE,(WI)
E0B3 E0
E0B4 B7         100          OR       A
E0B5 ED 52      101          SBC      HL,DE
E0B7 D8         102          RET      C
E0B8            103 ;        IF (G<0) THEN X=X+1
E0B8 2A 19 E0   104 IF2      LD       HL,(WG)
E0BB CB 7C      105          BIT      7,H
E0BD CA D2 E0   106          JP       Z,ELSE2
E0C0            107 ;                 G=G+Q2
E0C0 21 15 E0   108 THEN2    LD       HL,WX
E0C3 34         109          INC      (HL)
E0C4 2A 19 E0   110          LD       HL,(WG)
E0C7 ED 5B 13   111          LD       DE,(WQ2)
E0CA E0
E0CB 19         112          ADD      HL,DE
E0CC 22 19 E0   113          LD       (WG),HL
E0CF C3 E3 E0   114          JP       PSET1
E0D2            115 ;          ELSE Y=Y+1
E0D2 21 17 E0   116 ELSE2    LD       HL,WY
E0D5 34         117          INC      (HL)
E0D6            118 ;                 G=G-P2
E0D6 2A 19 E0   119          LD       HL,(WG)
E0D9 ED 5B 11   120          LD       DE,(WP2)
E0DC E0
E0DD B7         121          OR       A
E0DE ED 52      122          SBC      HL,DE
E0E0 22 19 E0   123          LD       (WG),HL
E0E3            124 ;     PSET(X,Y,C)
E0E3 CD 61 E0   125 PSET1    CALL     PSET
E0E6            126 ;     NEXT
E0E6 C3 A4 E0   127 NEXT1    JP       FOR1
E0E9            128
E0E9            129 ;      *)
E0E9            130 ;     ELSE
E0E9            131 ;     (*
E0E9            132 ;       Q=Y1-Y2
E0E9 2A 05 E0   133 ELSE1    LD       HL,(WY1)
E0EC ED 5B 09   134          LD       DE,(WY2)
E0EF E0
E0F0 B7         135          OR       A
E0F1 ED 52      136          SBC      HL,DE
E0F3 22 0F E0   137          LD       (WQ),HL
E0F6            138 ;       Q2=Q*2
E0F6 29         139          ADD      HL,HL
E0F7 22 13 E0   140          LD       (WQ2),HL
E0FA            141 ;       G=P2-Q
E0FA 2A 11 E0   142          LD       HL,(WP2)
E0FD ED 5B 0F   143          LD       DE,(WQ)
E100 E0
E101 B7         144          OR       A
E102 ED 52      145          SBC      HL,DE
E104 22 19 E0   146          LD       (WG),HL
E107            147 ;       FOR I=1 TO P+Q
E107 21 00 00   148          LD       HL,0
E10A 22 71 E0   149          LD       (WI),HL
E10D            150
E10D 21 71 E0   151 FOR3     LD       HL,WI
E110 34         152          INC      (HL)
E111            153
E111 2A 0D E0   154          LD       HL,(WP)
E114 ED 5B 0F   155          LD       DE,(WQ)
E117 E0
E118 19         156          ADD      HL,DE
E119            157 ;        LD       DE,HL
E119 ED 5B 71   158          LD       DE,(WI)
E11C E0
E11D B7         159          OR       A
E11E ED 52      160          SBC      HL,DE
E120 D8         161          RET      C
E121            162 ;        IF (G>0) THEN
E121 2A 19 E0   163 IF3      LD       HL,(WG)
E124 11 00 00   164          LD       DE,0
E127 B7         165          OR       A
E128 ED 52      166          SBC      HL,DE
E12A FA 41 E1   167          JP       M,ELSE3
E12D            168 ;                       (* X=X+1
E12D 21 15 E0   169 THEN3    LD       HL,WX
E130 34         170          INC      (HL)
E131            171 ;                          G=G-Q2
E131 2A 19 E0   172          LD       HL,(WG)
E134 ED 5B 13   173          LD       DE,(WQ2)
E137 E0
E138 B7         174          OR       A
E139 ED 52      175          SBC      HL,DE
E13B 22 19 E0   176          LD       (WG),HL
E13E C3 50 E1   177          JP       PSET3
E141            178 ;                        *)
E141            179 ;                     ELSE
E141            180 ;                       (* Y=Y-1
E141 21 17 E0   181 ELSE3    LD       HL,WY
E144 35         182          DEC      (HL)
E145            183 ;                          G=G+P2
E145 2A 19 E0   184          LD       HL,(WG)
E148 ED 5B 11   185          LD       DE,(WP2)
E14B E0
E14C 19         186          ADD      HL,DE
E14D 22 19 E0   187          LD       (WG),HL
E150            188 ;                        *)
E150            189 ;          PSET(X,Y,C)
E150 CD 61 E0   190 PSET3    CALL     PSET
E153            191 ;   NEXT
E153 C3 0D E1   192 NEXT3    JP       FOR3
E156            193 ;
E156            194 ; ENDFUNC
E156            195
```


▶私は以前，編集室へX1のMusic BASICの音色データのX-BASICへのコンバートの質問を出しましたが解答がなく，もうしかたないとあきらめています。ところが，1990年3月号の●●VEは」「となりのトトロ」の音色データがそのまま表記されていました。これだとX-BASICにもすぐ移せてよいと思います。　　小川　謙治(17) X68000ACE　島根県

デバイスドライバを作る(後)

Murata Toshiyuki 村田 敏幸

後編は割り込みルーチンの話とサンプルプログラムの作成です。理論中心の前編に比べてより実践的な内容になっています。サンプルを参考にすれば自分だけのオリジナルデバイスドライバを作ることもできます。

さて，後編だ。さっそく，Human68kがデバイスドライバをどういった手順で呼び出すかという話から始めよう。

デバイスドライバの呼び出し手順

Human68k←→デバイスドライバ間のデータの受け渡しはリクエストヘッダ（Request header）と呼ばれるメモリブロックを介して行われる[1]。デバイスドライバを呼び出す際に，Human68kは処理内容を格納したリクエストヘッダを作成し，その先頭アドレスをa5レジスタに入れてデバイスドライバに渡す。リクエストヘッダの構造や大きさは処理内容ごとに異なるが，先頭の5バイトは表1のように決まっている。入力を行うのか出力なのかといった処理の種類は0～12のコマンドコード（表2）で表され，リクエストヘッダの2バイト目に格納される。

デバイスドライバ側は渡されたリクエストヘッダからコマンドコードや付随するパラメータを取り出し，必要な処理を行ってから，処理がうまくいったかどうかというステータスをリクエストヘッダに格納して返す。終了ステータスは正常終了時は0000_H，エラーのときは図1および表3に示す2バイトのエラーコードで，リクエストヘッダの3，4バイト目に下位バイト，上位バイトの順でセットする[2]。デバイスドライバがエラーを返した場合は，Human68kは即座にエラー処理に入り，画面中央にエラーメッセージを表示し，中止/再実行/無視の選択をユーザーに求める。中止/再実行/無視の選択肢のうちどれを有効とするかはデバイスドライバの返すエラーコードで指定することができる。

上の説明ではHuman68kがリクエストヘッダをデバイスドライバに渡すと一気に処理を済ませてしまうような書き方をしたが，現実にはデバイスドライバの呼び出しは次のように2度に分けて行われる。

1）Human68kはリクエストヘッダを作成し，その先頭アドレスをa5レジスタに入れて，デバイスドライバのストラテジエントリをコールする。

2）デバイスドライバのストラテジルーチンでは渡されたリクエストヘッダへのポインタを内部に待避し，いったんHuman68kに戻る。

3）Human68kはすかさずデバイスドライバの割り込みルーチンを呼び出す。

4）インタラプトルーチンでは2)で待避しておいたリクエストヘッダへのポインタを改めて取り出し，実際の入出力処理を行う。

このような奇妙な手順になっているのは，将来Human68kがマルチタスク化されるときに対応しやすくするため，とのことだ[3]。マルチタスクのシステムでは，並行して走っている[4]複数の処理単位(呼び方はシステムによってさまざまだが，ここでは“タスク”と呼ぼう）がデバイスドライバの都合を考えずに入出力を要求してくる。当然，デバイスドライバが処理を行っている最中に別のタスクが同一のデバイスドライバに入出力を要求することもある。さすがに複数の入出力要求を同時に満たすことはできな

1）Human68kがなにかにつけ参考にしているMS-DOSでは，Human68kでいうリクエストヘッダのことを単にコマンドパケット（Command packet）と呼び，特にその先頭の部分のことをリクエストヘッダと呼んでいる。

2）バイト順がひっくり返っているのはMS-DOSに合わせたためらしい。こんなことまで真似しなくてもよかったように思うが。

3）もっとも，マルチタスク化されたときにいままでのデバイスドライバがそのまま使えるというわけではないようだ。

4）“並行して走っている”とはいってもCPUがひとつしかない限り，ある瞬間にはひとつのプログラムしか動かしようがない。細かく時間を分割し，各タスクを少しずつ実行することで見かけ上並行動作しているように見せるわけだ。

表1 リクエストヘッダの先頭5バイト

0	26という定数
1	ブロックデバイスではユニット番号 キャラクタデバイスでは未使用
2	コマンドコード
3	終了ステータス下位バイト
4	終了ステータス上位バイト

表2 コマンドコード

コード	キャラクタデバイス	ブロックデバイス
0	初期化	初期化
1		ディスク交換チェック
2		
3	IOCTRLによる入力	IOCTRLによる入力
4	入力	入力
5	先読み入力	ドライブのコントロールおよびセンス
6	入力ステータスチェック	
7	入力バッファクリア	
8	出力（VERIFY OFF時）	出力（VERIFY OFF時）
9	出力（VERIFY ON時）	出力（VERIFY ON時）
10	出力ステータスチェック	
11		
12	IOCTRLによる出力	IOCTRLによる出力

▶ついに買いました。KORG TZ！ 音の良さに感動しています。メーカーさん，ぜひMIDI対応ゲームをM1&Tシリーズにも対応させてください。
稲富 顕二(18) X68000 岡山県

いし，といって要求があるたびにその場その場で入出力処理を行ってしまおうなんて考えるとシステム全体の効率が悲惨なことになる。結局，入出力要求をどこかに溜め込んでおき，そこからひとつずつ取り出しては処理するような細工が必要になってくる。その管理をOSに任せることも考えられないではないが，結構負担が大きいのでここはデバイスドライバ側で対応する。入出力要求を受け取るだけのルーチンと，入出力処理そのものを行うルーチンを分離しておくのだ。前者は入出力要求があるごとに呼び出され，要求された仕事をキュー（Queue：待ち行列）に加えて，すぐOSに戻る。後者はタスク管理の一環として定期的に呼び出され，キューから入出力要求を取り出しては実際の入出力処理を行う。

で，一応将来のそういう使い方を想定して，いまのうちからHuman68kのデバイスドライバには2つのエントリポイントが設けられているわけだ。ただし，Human68kが本当にマルチタスク化されるかどうかはまだわからない。

サンプルで試してみよう

Human68kのデバイスドライバを構成するデバイスヘッダ，ストラテジルーチン，割り込みルーチンの3つのモジュールのうち，デバイスヘッダに関しては先月話してある。また，ストラテジルーチンは前述のようにa5で渡されるリクエストヘッダを内部ワークに待避して戻るだけのルーチンであり，

```
move.l  a5,待避用ワーク
rts
```

の2行で書けてしまう。あとはデバイスドライバの実質的な本体である割り込みルーチンが残っているわけだが，先に これまでの話を確認する意味で，ごく簡単なサンプルプログラムを見てもらおう。

リスト1のCTESTDRV.Sは非常に単純なキャラクタデバイスドライバのサンプルだ。先月のDRIVER.Hをインクルードしていることに注意して，

```
AS CTESTDRV
LK CTESTDRV
REN CTESTDRV.X CTESTDRV.SYS
```

のようにアセンブル/リンクしたえうで“.SYS”にリネームするか，そうでなければ，

```
LK CTESTDRV /OCTESTDRV.SYS
```

とリンク時にOスイッチでファイル名を指定してCTESTDRV.SYSを作成する[5]。その後CONFIG.SYSに，

```
DEVICE = ［パス名］CTESTDRV.SYS
```

の1行を追加してからシステムを立ち上げ直せばCHRTESTというデバイスが使えるようになる[6]。このデバイスCHRTESTは入力時には無条件にEOFコード($1A_H$)を返し，出力時には出力データをどこにも送らずに捨ててしまう“何もしないデバイス”だ。

では，リスト1の頭の部分から見ていこう。プログラムの先頭，12行からデバイスヘッダが始まっている。リンクポインタはデバイスドライバリンクの終わりを示す－1になっているが，組み込み時にHuman68kによって次のデバイスドライバを指すように勝手に書き換えられる。デバイス属性は，キャラクタデバイスで，IOCTRLは不可で，COOKEDモード，という属性を与えてある。記号定数を使って書いてあるから一目瞭然だろう[7]。それから，ストラテジルーチンへのポインタ，割り込みルーチンへのポインタ，デバイス名が順に並んでいる。デバイス名は8バイト必要なので，CHRTESTの7文字の後ろにスペースが1個加えてある。

26行からのストラテジルーチンでは，20行で用意したワークにリクエストヘッダへのポインタであるa5を格納している。そして，その後ろが肝心の割り込みルーチンだ。先月から続くこの話もようやく佳境にはいる。34行以下では割り込みルーチンの基本形が示されている。最初にデバイスドライバ内で使用するレジスタをスタックに待避する(35行)。デバイスドライバ中ではレジスタの値を保存しておかなければならないのだ[8]。リスト1では使用するレジスタであるd0とa4,a5だけを待避しているが，いちいちレジスタの使用状況を調べるのが面倒であれば，d0～d7とa0～a6の全レジスタを待避してしまえばよい。ただし，movemは複数のレジスタを1命令で転送できるとはいえ，転送するレジスタの数が多いと実行に時間がかかるので，速度を気にするのであれば無駄なレジスタ待避はしないほうがいい。

5) デバイスドライバの拡張子はべつに“.SYS”である必要はなく，CONFIG.SYSに登録する名前とつじつまが合っていればなんでもかまわない。が，誤って実行してしまわないよう，“.X”のままにしておくのだけはやめておいたほうがよいだろう。

6) キャラクタデバイスドライバを作るときに，ソースファイル，実行ファイルのファイル名はデバイス名と同じにしてしまってはいけない。開発中はいいが，いざプログラムが完成し組み込んだ時点で，デバイス名と同名のファイルは読み書きできなくなってしまうからだ。

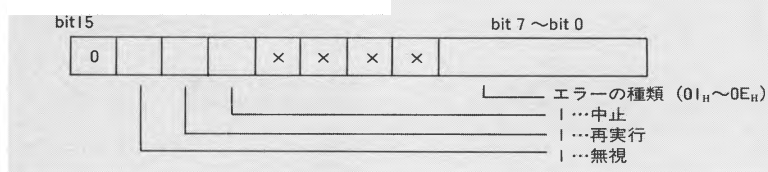
7) ここで使っている定数はDRIVER.Hの中で定義されている。

8) ccrまでは気にしなくてよい。

表3 エラーの種類

コード	内容
01_H	ユニット番号が不正
02_H	ドライブの準備ができていない（ディスクが入っていない）
03_H	コマンドコードが無効
04_H	CRCエラー
05_H	ディスクの管理領域が破壊されている
06_H	シークエラー
07_H	メディアが無効
08_H	指定セクタが見つからない
09_H	プリンタオフライン
$0A_H$	書き込みエラー
$0B_H$	読み込みエラー
$0C_H$	そのほかのエラー
$0D_H$	ライトプロテクト（ディスクの取り換え不可）
$0E_H$	ライトプロテクト（ディスクの取り換え可）

図1 エラーコード

▶ナイトアームズに出てくる女性型ロボットはARIELにそっくりだ！ と思っているのは私だけではないはずだ。 野村 幸司 (19) X68000, X1 広島県

続いて37行でストラテジルーチンでしまっておいたリクエストヘッダへのポインタをa5に取り出している。それからコマンドコードに応じて処理を振り分ける。各コマンドの処理は80行以下にサブルーチンの形で用意してある。ここではジャンプテーブルの手法によって多方向への分岐をすっきりとまとめている。61行以下が各コマンドコードに対応した処理ルーチンの先頭アドレスを順に並べたジャンプテーブルだ。68000のアドレスは4バイトを占めるから，コマンドコードを4倍し，ジャンプテーブルの先頭アドレスに加え，“コマンドの処理ルーチンの先頭アドレスを格納しているジャンプテーブル上の位置”を求める（39～44行）。そこから処理ルーチンの先頭アドレスをレジスタに取り出し(46行)，そのアドレスをサブルーチンコールしている(48行)。bsrでは分岐先にラベルしか指定できないので，アドレッシングモードの豊富なjsr命令を使った。48行の，

```
jsr    (a4)
```

は“a4の指すアドレスをコールする”という意味だ。

処理が済んでサブルーチンから戻ってきたら，終了ステータスをリクエストヘッダ内に格納する。各コマンドの処理ルーチンは終了ステータスをd0.wに入れて戻るように作ってあるので，51行でd0.wの下位バイトを，続いて8ビット左シフトして53行で上位バイトをセットしている。あとは待避してあったレジスタを復帰してrtsでHuman68kに戻る。

割り込みルーチンの話

では，割り込みルーチンがサポートする各コマンドについて順に解説していこう。なお，本来ならキャラクタデバイスとブロックデバイスの両方を網羅

リスト1　CTESTDRV.S

```
  1: *        実験用デバイスドライバ
  2:
  3:          .include        doscall.mac
  4:          .include        const.h
  5:          .include        driver.h
  6: *
  7:          .text
  8:          .even
  9: *
 10: *        デバイスヘッダ
 11: *
 12: device_header:
 13:          .dc.l   -1
 14:          .dc.w   CHAR_DEVICE+DISABLE_IOCTRL+COOKED_MODE
 15:          .dc.l   strategy_entry
 16:          .dc.l   interrupt_entry
 17:          .dc.b   'CHRTEST '
 18: *                 12345678
 19: *
 20: request_header:                       *リクエストヘッダ待避領域
 21:          .dc.l   0
 22:
 23: *
 24: *        ストラテジルーチン
 25: *
 26: strategy_entry:
 27:          move.l  a5,request_header        *リクエストヘッダへのポインタを
 28:                                           *  待避して
 29:          rts                              *速やかに戻る
 30:
 31: *
 32: *        割り込みルーチン
 33: *
 34: interrupt_entry:
 35:          movem.l d0/a4-a5,-(sp)           *レジスタ待避
 36:
 37:          movea.l request_header,a5        *a5=リクエストヘッダ
 38:
 39:          moveq.l #0,d0                    *d0.l=コマンドコード
 40:          move.b  CMD_CODE(a5),d0          *
 41:          add.w   d0,d0                    *2倍する
 42:          add.w   d0,d0                    *2倍の2倍で4倍
 43:          lea.l   jump_table,a4            *a4=ジャンプテーブル先頭
 44:          adda.l  d0,a4                    *a4=コマンド処理ルーチンへの
 45:                                           *  ポインタへのポインタ
 46:          movea.l (a4),a4                  *a4=コマンド処理ルーチンへの
 47:                                           *  ポインタ
 48:          jsr     (a4)                     *a4の指すアドレスを
 49:                                           *  サブルーチンコール
 50:
 51:          move.b  d0,ERR_LOW(a5)           *終了ステータスをセット
 52:          lsr.w   #8,d0                    *
 53:          move.b  d0,ERR_HIGH(a5)          *
 54:
 55:          movem.l (sp)+,d0/a4-a5           *レジスタ復帰
 56:          rts                              *Humanへ戻る
 57:
 58: *
 59: *        コマンド処理ジャンプテーブル
 60: *
 61: jump_table:
 62:          .dc.l   init            *0       初期化
 63:          .dc.l   notcmd          *1       (無効)
 64:          .dc.l   notcmd          *2       (無効)
 65:          .dc.l   ioctrl_in       *3       IOCTRLによる入力
 66:          .dc.l   input           *4       入力
 67:          .dc.l   sense           *5       1バイト先読み入力
 68:          .dc.l   inpstat         *6       入力ステータスをチェック
 69:          .dc.l   flush           *7       入力バッファをクリア
 70:          .dc.l   output          *8       出力(VERIFY OFF)
 71:          .dc.l   voutput         *9       出力(VERIFY ON)
 72:          .dc.l   outstat         *10      出力ステータスをチェック
 73:          .dc.l   notcmd          *11      (無効)
 74:          .dc.l   ioctrl_out      *12      IOCTRLによる出力
 75:
 76: *
 77: *        各コマンドの処理
 78: *
 79:
 80: *
 81: *        無効(コマンドコード1,2,11)
 82: *        IOCTRLによる入力(コマンドコード3)
 83: *        IOCTRLによる出力(コマンドコード12)
 84: *
 85: notcmd:
 86: ioctrl_in:
 87: ioctrl_out:
 88:          move.w  #ILLEGAL_CMD,d0          *エラーコードを持って
 89:          rts                              *  戻る
 90:
 91: *
 92: *        入力(コマンドコード4)
 93: *
 94: input:
 95:          tst.l   DMA_LEN(a5)              *入力要求が0バイトであれば
 96:          beq     done                     *  何もせずに戻る
 97:                                    *そうでなければ
 98:          movea.l DMA_ADR(a5),a4           *a4=データ読み込み領域
 99:          move.b  #EOF,(a4)                *入力データをセット
100:
101: done:    moveq.l #0,d0                    *正常終了
102:          rts
103:
104: *
105: *        1バイト先読み入力(コマンドコード5)
106: *
107: sense:
108:          move.b  #EOF,SNS_DATA(a5)        *先読みデータをセット
109:          bra     done                     *正常終了
110:
111: *
112: inpstat:                    *入力ステータスチェック(コマンドコード6)
113: flush:                      *入力バッファクリア(コマンドコード7)
114: output:                     *VERIFY OFF時の出力(コマンドコード8)
115: voutput:                    *VERIFY ON時の出力(コマンドコード9)
116: outstat:                    *出力ステータスチェック(コマンドコード10)
117:          bra     done                     *正常終了
118: *
119: *        ↑ここまでがメモリに居坐るデバイスドライバ本体
120:
121: *
122: *        初期化(コマンドコード0)
123: *
124: init:
125:          pea.l   title                    *タイトルを表示
126:          DOS     _PRINT                   *
127:          addq.l  #4,sp                    *
128:
129:          move.l  #init,DEV_END_ADR(a5)    *デバイスドライバ本体の
130:                                           *  最終アドレスをセット
131:
132:          bra     done                     *正常終了
133: *
134:          .data
135:          .even
136: *
137: title:                                    *タイトルメッセージ
138:          .dc.b   CR,LF
139:          .dc.b   '実験用キャラクタデバイス',CR,LF
140:          .dc.b   'CHRTESTの名前で入出力試験が行えます',CR,LF
141:          .dc.b   0
142:
143:          .end
```

▶こんにちは！　覚えていますか？　去年のSTUDIOXに載った村上です。この1年間パソコンを封印……しなかったり，試験1カ月前にメガドライブを買ったり(！)したけど，大学生になれました。不思議ですねー。TZR50に乗っている僕を見掛けたら声をかけて！
村上　浩二(19)X1G　広島県

すべきなのだろうが，ここではキャラクタデバイスの場合のみを取り上げる。また，CTESTDRV.Sはあまりに簡略化しすぎたためあまりよい例にはなっていない。それでも，ないよりはマシなので，適当にリストを参照しながら読み進めてもらいたい。

●初期化

入力： 2(a5) 1B コマンドコード（0）
18(a5) 1L パラメータへのポインタ
出力： 3(a5) 1B 終了ステータス（下位）
4(a5) 1B 終了ステータス（上位）
14(a5) 1L デバイスドライバで使用するメモリの最終アドレス+1

初期化ルーチンはデバイスドライバ組み込み時にただ一度だけ呼び出される。CONFIG.SYSに記述されたパラメータを受け取り，必要に応じて装置などの初期化を行ったのち，デバイスドライバで使用するメモリの最終アドレス+1をリクエストヘッダ内にセットして戻る。パラメータは“DEVICE =～”行の“=”の直後を指すポインタの形で渡される。ただし，“.X”ファイルのコマンドライン文字列とは異なり，パラメータはあらかじめHuman68kによって語単位で区切られている。CONFIG.SYSに，

DEVICE = A:¥SYS¥RAMDISK.SYS #G

と記述してあったとすると，パラメータは，

```
.dc.b  'A:¥SYS¥RAMDISK.SYS',0
.dc.b  '#G',0
.dc.b  0
```

のように分解される。空白を0に置き換え，最後にもうひとつ0を付け加えた形だ[9]。

9）空白はいくつあっても1個の0に置き換えられる。

通常，初期化ルーチンはデバイスドライバプログラムの最後に置き，初期化後は切り離す。つまり，OSに戻るときに初期化ルーチンの先頭アドレスを14（a5）にセットするわけだ。このとき，以後のデバイスドライバの動作に必要なデータまで誤って切り離さないよう十分注意しなければならない。メモリ上のプログラムは常にテキストセクション，データセクション，ブロックトレースセクションの順に並んでいるから，テキストセクションにある初期化ルーチン以降を切り離すとデータセクションやブロックトレースセクションも一緒になくなってしまう。デバイスドライバを作成する際にはデータ類も初期化ルーチンより前のテキストセクションに置いてお

リスト2 CTESTDRV.S追加部

```
  1: *
  2: *          試験用ルーチン
  3: *
  4: test:
  5:            move.l  d1,-(sp)
  6:
  7:            bsr     showcmd                     *コマンドの種類を表示
  8:            bsr     showlen                     *入出力系コマンドであれば
  9:                                                *  データ長を表示
 10:            move.l  (sp)+,d1
 11:            rts
 12:
 13: *
 14: *          コマンドの種類を表示する
 15: *
 16: showcmd:
 17:            moveq.l #0,d0                       *コマンドの種類を表す文字列の
 18:            move.b  CMD_CODE(a5),d0             *  先頭アドレスをa4に得る
 19:            add.w   d0,d0                       *
 20:            add.w   d0,d0                       *
 21:            lea.l   cmd_table,a4                *
 22:            add.l   d0,a4                       *
 23:            movea.l (a4),a4                     *
 24:
 25:            move.l  a4,-(sp)                    *コマンドの種類を表示する
 26:            move.w  #1,-(sp)                    *
 27:            DOS     _CONCTRL                    *
 28:
 29:            move.l  #crlfms,2(sp)               *改行する
 30:            DOS     _CONCTRL                    *
 31:            addq.l  #6,sp                       *
 32:            rts
 33:
 34: *
 35: *          メッセージへのポインタのテーブル
 36: *
 37: cmd_table:
 38:            .dc.l   mes00,mes01,mes02,mes03
 39:            .dc.l   mes04,mes05,mes06,mes07
 40:            .dc.l   mes08,mes09,mes10,mes11
 41:            .dc.l   mes12
 42:
 43: *
 44: *          入出力系コマンドであればデータ長を16進表示する
 45: *
 46: showlen:
 47:            moveq.l #0,d0                       *d0.l=コマンド番号
 48:            move.b  CMD_CODE(a5),d0             *
 49:            move.l  #%00010011_00011000,d1      *入出力系コマンドかどうかを
 50:            btst.l  d0,d1                       *  調べる
 51:            beq     slen0                       *そうでなければ何もしない
 52:
 53:            pea.l   temp                        *データ長を16進8桁に変換する
 54:            move.l  DMA_LEN(a5),-(sp)           *
 55:            bsr     itoh                        *
 56:            addq.l  #8,sp                       *
 57:            pea.l   temp                        *表示する
 58:            move.w  #1,-(sp)                    *
 59:            DOS     _CONCTRL                    *
 60:
 61:            move.l  #crlfms,2(sp)               *改行する
 62:            DOS     _CONCTRL                    *
 63:            addq.l  #6,sp                       *
 64: slen0:     rts
 65:
 66: *
 67: *          数値→16進文字列変換
 68: *
 69: value      =       8
 70: buff       =       12
 71: *
 72: itoh:
 73:            link    a6,#0
 74:            movem.l d0-d2/a0,-(sp)
 75:
 76:            move.l  value(a6),d0        *値
 77:            movea.l buff(a6),a0         *文字列格納アドレス
 78:
 79:            moveq.l #8-1,d2             *以下を8回繰り返す
 80:
 81: itoh0:     rol.l   #4,d0               *d0.lを左に4ビット回転する
 82:            move.b  d0,d1               *d0の下位バイトをd1に取り出し
 83:            andi.b  #$0f,d1             *  下位4ビットを残してマスクする
 84:            addi.b  #'0',d1             *ここで数値から16進を表す文字へ
 85:            cmpi.b  #'9'+1,d1           *  変換する
 86:            bcs     itoh1               *  0～9の場合は'0'を足すだけだが
 87:            addq.b  #'A'-'0'-10,d1      *   A～Fの場合はさらに補正が必要
 88:
 89: itoh1:     move.b  d1,(a0)+            *変換した文字をしまう
 90:
 91:            dbra    d2,itoh0            *繰り返す
 92:
 93:            clr.b   (a0)                *文字列終端コードを書き込む
 94:
 95:            movem.l (sp)+,d0-d2/a0
 96:            unlk    a6
 97:            rts
 98:
 99: *
100: *          コマンドの種類表示用文字列
101: *
102: mes00:     .dc.b   '初期化',0
103: mes01:     .dc.b   'コマンド1（無効）',0
104: mes02:     .dc.b   'コマンド2（無効）',0
105: mes03:     .dc.b   'IOCTRLによる入力',0
106: mes04:     .dc.b   '入力',0
107: mes05:     .dc.b   '先読み入力',0
108: mes06:     .dc.b   '入力ステータスをチェック',0
109: mes07:     .dc.b   '入力バッファをクリア',0
110: mes08:     .dc.b   '出力（VERIFY OFF）',0
111: mes09:     .dc.b   '出力（VERIFY ON）',0
112: mes10:     .dc.b   '出力ステータスをチェック',0
113: mes11:     .dc.b   'コマンド11（無効）',0
114: mes12:     .dc.b   'IOCTRLによる出力',0
115: *
116: crlfms:    .dc.b   CR,LF,0                             *改行用文字列
117: *
118: temp:      .ds.b   8+1                                 *16進変換用バッファ
119: *
120:            .even
```

かなければならないことになる。

●入力（コマンドコード4）

入力： 2(a5)　1B　コマンドコード（4）
　　　14(a5)　1L　転送先バッファへのポインタ
　　　18(a5)　1L　入力要求バイト数
出力： 3(a5)　1B　終了ステータス（下位）
　　　4(a5)　1B　終了ステータス（上位）

装置から指定されただけのバイト数を読み込み，指定バッファに順に書き込む。データが用意できていない場合は揃うまで待つ[10]。なお，COOKEDモードでは入力要求バイト数は常に1になるはずだが，先月話したようにCOOKEDモードとRAWモードはioctrlで切り換えることができるため，デバイス属性がCOOKEDでもRAWモードでアクセスされる可能性がある[11]。また，入力要求バイト数が0ということもありうるようだ。よって，デバイスドライバの入力処理ルーチンは入力要求バイト数が何バイトであろうとも過不足なく処理が行えるよう作る必要がある。

●先読み入力（コマンドコード5）

入力： 2(a5)　1B　コマンドコード（5）
出力： 3(a5)　1B　終了ステータス（下位）
　　　4(a5)　1B　終了ステータス（上位）
　　　13(a5)　1B　先読みしたデータ

装置から1バイト先読みし，リクエストヘッダ内にセットして戻る。先読みだから，読み込んだデータはあとでコマンドコード4で入力される場合に備えて残しておく。もしデータがない場合はデータが揃うのを待ったりせず，即座に0を返す。

●入出力ステータスチェック（コマンドコード6,10）

入力： 2(a5)　1B　コマンドコード（6,10）
出力： 3(a5)　1B　終了ステータス（下位）
　　　4(a5)　1B　終了ステータス（上位）

装置が現在入力可能かどうか（コマンドコード6），出力可能かどうか（コマンドコード10）を調べ，可能であれば0000_Hを，不可能であれば0001_Hを終了ステータスとして返す。正常終了時に0000_H以外の終了ステータスを返す唯一の例外だ。

●入力バッファクリア（コマンドコード7）

入力： 2(a5)　1B　コマンドコード（7）
出力： 3(a5)　1B　終了ステータス（下位）
　　　4(a5)　1B　終了ステータス（上位）

入力データを一度内部のバッファに溜め込み，入力要求があった場合はそのバッファからデータを取り出して返すような構成のデバイスドライバにおいて，そのバッファに溜まっていたデータを破棄するコマンドだ。コマンドコード5で先読みしていたデータが保存されているときには一緒に捨てる。

●出力（コマンドコード8,9）

入力： 2(a5)　1B　コマンドコード（8,9）
　　　14(a5)　1L　出力データへのポインタ
　　　18(a5)　1L　出力要求バイト数
出力： 3(a5)　1B　終了ステータス（下位）
　　　4(a5)　1B　終了ステータス（上位）

指定アドレス以降のデータを指定されただけのバイト数分装置に出力する。コマンドコード9の場合は，出力したデータをすかさず読み込み，出力前のデータと比較して正しく出力できているかどうかを確認する（一致しない場合は書き込みエラーを返す）。入力コマンドのところでしたCOOKEDモードとRAWモードに関する注意はそのまま当てはまる。

●IOCTRLによる入出力（コマンドコード3,12）

リクエストヘッダの構成はそれぞれ，コマンドコード4，コマンドコード8と同様なので省略する。この機能をサポートしない場合は無効なコマンドコードとしてきちんとはじかなければならない。

入出力コマンドはともかく，そのほかのコマンドがどのような状況で呼び出されるかというのはなかなか興味深い問題だ。そこで，ものは試し，CTESTDRV.Sに与えられたコマンドを表示する試験用ルーチンを追加してみよう。リスト1の38行に，

```
bsr test
```

の1行を付け加え，リスト2を初期化ルーチンより前，118行あたりにでも挿入し，再アセンブルしてもらいたい。これにより，CHRTESTが呼び出されるたびに，画面にコマンドの種類が表示されるようになる。ついでに入出力系のコマンドの場合は入出力要求されたバイト数も一緒に表示するようにしてある。本来文字表示を行うはずのないタイミングで強引に表示を行っていることもあり，画面がかなりうるさくなるし，ときに文字化けを起こすこともあるが，試験用ということで勘弁してもらいたい。

```
TYPE CONFIG.SYS > CHRTEST
```

なんてやってみると，1文字ごとにデバイスドライバが呼び出される様子を見ることができる。

```
COPY CHRTEST TEMPFILE
```

とか，

```
UPPER CHRTEST
```

とかいろいろ試してみるとデバイスドライバの動作がよりよくつかめるだろう。デバイス属性をRAWモードに変更してアセンブルし直し，COOKEDモードとRAWモードの違いを確認するのも面白い。

一応プログラムについても触れておこう。コマンド名の表示は4行以下のサブルーチンshowcmdで行っている。コマンド名を表す文字列へのポインタをテーブルにしておき，ジャンプテーブルのときとほとんど同じ手順で文字列へのポインタを取り出し，DOSコールconctrlで表示している。printではなく

10)　たとえば，入力先デバイスがキーボードであるならキーが押されるまで，RS-232Cポートならデータが送信されてくるまで待つわけだ。
11)　もっとも，キャラクタデバイスからRAWモードで入力するのには先月話したようないくつかの問題があり，COOKEDモードの入力デバイスをRAWモードに切り換えて使うことはまずないといってよいが。

▶オタクじゃない！　オタクじゃない！　と，ガンダムZZのオープニングテーマの替え歌を歌う私は，やはりオタクであろうか……。　松前　龍次（17）X1twin　山口県

conctrlを使っているのは，

TYPE CONFIG.SYS > CHRTEST

のようにリダイレクトされた場合に備えてのことだ。リダイレクトすることで標準出力にCHRTESTが割り付けられ，そのCHRTESTの処理ルーチンの中からさらに標準出力に出力すると，自分自身を呼び出すわけだから無限ループに陥ってしまう。

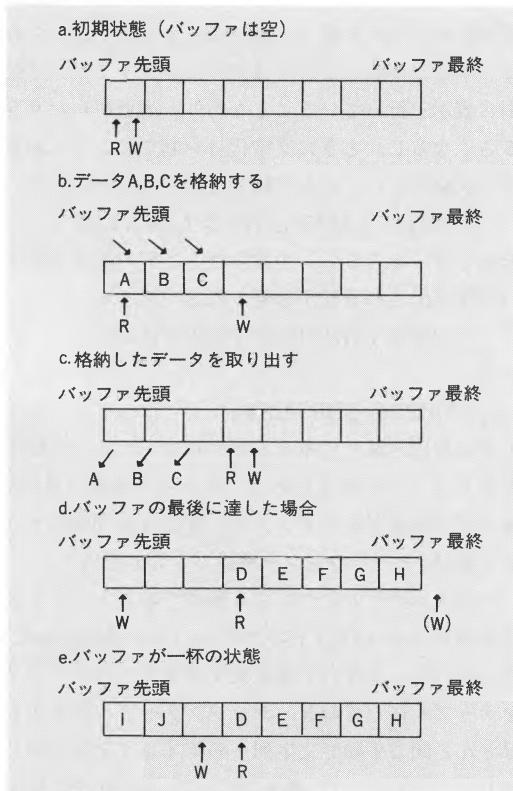
プログラム上のテクニックとしては47〜51行をチェックしておきたい。ここではコマンドコードが入出力系のコマンド（3，4，8，9，12）であるかどうかをbtst命令を使って調べている。d1に第3，4，8，9，12ビットだけが1であるようなデータを，d0にコマンドコードを入れておいて，

btst.l　d0, d1

を実行すれば，結果のZビットだけでd0が上記5つの値のどれかと一致しているかどうかがわかるという寸法だ。cmpを5個並べるよりはずーっとスマートだと思うがどうだろうか。

実用的サンプルプログラム

CTESTDRV.Sがあまり参考にならなかったので，もうひとつ，今度はもっと実用になるデバイスドライバを作ってみよう。次のような動作をするキャラクタデバイスを考える。

1)　出力時には出力データを画面に表示しながら，同時に内部のバッファにも溜め込む。

2)　入力時には1)で溜め込んだデータを返す。

知っている人は知っているように，これは祝社長の手により電脳倶楽部のVol.1を飾ったプログラムと同機能だ。なかなか便利だし，プログラムの題材としても手頃なのでアイデアを拝借してきた。オリジナルに敬意を表してデバイス名も同名のTAPにしてしまおう。さっきのCTESTDRV.Sにぽこぽことコマンド処理ルーチンを加えていったらリスト3のように仕上がった。なお，オリジナルではバッファの大きさをCONFIG.SYS中で指定できるようになっているのだが，リスト3では固定されている。サイズを変更したいときは201行の定数を加減してアセンブルし直すことになる。

さて，このプログラムではデバイスドライバとしての構造うんぬんよりも，データを（あとでちゃんと順序よく取り出せるような形で）溜め込む処理そのものが重要なポイントだ。こういった場合はリングバッファというデータ構造を使うのが定石になっている。リングバッファはFIFO（First In First Out：最初に入れたものが最初に出てくる）を実現するデータ構造で，キューを形成するのによく用いられる。新たに格納されるデータはバッファ内のデータ列末尾に付け加えられ，取り出すときは列の先頭のデータが取り出される。ちなみに，スタックは最後に入れたデータが最初に出てくるLIFO（Last In First Out）のデータ構造だ。

スタックはただ1個のポインタで管理されるが，リングバッファの場合はデータを書き込む位置を示すポインタと，データを読み出す位置を表すポインタの2つを使う。データがなにもない状態では，図2-aにあるように書き込み位置を表すポインタ（図ではWで表した）と読み出し位置を表すポインタ（同じくRで表した）は同じ位置を指している。ここから書き込み位置を進めつつデータを書き込んでいくと図2-bのようになる。WとRの差がバッファ内のデータの個数を表している。読み出すときには，読み出し位置のポインタのほうを進めながらデータを取り出していく。RがWに追いついたときにはバッファは空になっている（図2-c）。こうやって読み書きを行っていくと，いつかはポインタがバッファの最後を越えてしまうことになるが，そうなったらすかさずバッファ先頭を指すように修正してやる。バッファの最後と最初が論理的には繋がって輪になっていると考えるわけだ（図2-d）。また，読み出しを行わずに書き込みだけを続けるとRより先を指していたはずのWがぐるっと回ってRに追いつくことになる。完全にRとWが一致してしまうとバッファが空の状態と区別できなくなるので，その直前の状態（図2-e）をもってバッファが一杯であることを表

図2　リングバッファ

すと考える。この場合，バッファに格納できるデータの最大数はバッファの大きさよりデータ1個分少なくなる。それがもったいないと思うなら多少プログラムが複雑になるが，バッファ内のデータの個数を数えるカウンタを設ければバッファの大きさ一杯まで使えるようにできる。

と，ここまでが一般的なリングバッファの考え方だが，これをTAPドライバに採用するするにあたっては2点ほど考えておかなければならないことがある。ひとつはバッファが空のときに入力が要求されたときの処理だ。デバイスドライバの仕様ではデータがない場合はデータが入力されるのを待つことになっているが，TAPでは入力コマンドの処理中はどんなに待ってもデータは入ってくるはずがない。そこで，バッファが空の場合はEOFコード(CTRL-Z)を返すことにする。

もうひとつはバッファが一杯のときに出力が要求された場合だ。対策としては，

1) 即座にエラーを返す。
2) 新しいデータはバッファに追加せずに戻る。
3) 古いデータを消して新しいデータを格納する。

といった手段が考えられる。しかし，どれも完全とはいい難い。1)はある意味で正しいデバイスドライバの作り方だと思うが，入出力を要求したプログラムを止めなければならなくなる。2),3) はプログラムは止まらない代わりにバッファ内のデータが失われる。結局，リスト3では3) の方法を採用した。

では，軽くプログラムの解説をしていこう。まず，206行以下の初期化部分。ここではタイトルメッセージの表示とデバイスドライバで使用するメモリの最後をリクエストヘッダに格納しているだけで，CTESTDRV.Sと実質的な違いはない。ただし，初期化部を単に切り離すのではなく，ついでにリングバッファの分のメモリを確保している。199行のラベルbufftopがバッファの先頭を表しており，ここから定数BUFFSIZEを足した位置までをバッファとして使用する。バッファを.dsで確保せずに，初期化ルーチンと重ねてあるのがポイントだ。いつものようにブロックトレースセクションにバッファを.dsで確保したとすると，デバイスドライバ本体とバッファで初期化ルーチンを挟む形になり，初期化ルーチンを切り離すことができなくなる。といって，初期化ルーチンの前にバッファを確保しようとすると，textセクションに.dsを書かなければならず，実行ファイルがバッファの大きさの分膨れ上がってしまう。

```
        .text
        デバイスドライバ本体
        .bss
buff:   .ds.b   16*1024
        .text
        初期化ルーチン
```

のようにソース上で順序を変えてみたところで，アセンブル/リンク時に各セクションはひとつにまとめられてしまうので無駄だ。

94行以下が入力処理ルーチンだ。a0をバッファ内の読み出し位置を指すポインタ，a4を転送先へのポインタ，d2.lをループカウンタとして使用している。基本的には，

```
        move.b  (a0)+,(a4)+
```

を指定された入力バイト数回繰り返しているだけだが，バッファが空の場合にループを抜けてEOFコードを返す処理，および，ポインタがバッファの最後を越えたらバッファ先頭を指すように修正する処理が挟まっている。

また，123行からが出力ルーチンだ。ちょっと手を抜いてVERIFY ONのときの処理とOFFのときの処理を共通にし，ベリファイを省略してしまったが，それほど大きな問題にはならないだろう。バッファが一杯になったときに古いデータから消していく処理は，書き込み用のポインタが読み出し用のポインタに追いついたら，強制的に読み出し用ポインタを1バイト進めることで対処している。142行以下のCONCTRLの呼び出し方は多少姑息かもしれない。

乗算

除算命令divu，divsは以前紹介したし，すでに何度か使ってもいる。ここではこれらと対になる乗算命令mulu，mulsを紹介する。

mulu，mulsは

```
        mulu.w  #10,d0
        mulu.w  d1,d0
```

のようにして使い，データレジスタの下位ワードに16ビット数を掛け，結果を32ビットで求める命令だ（サイズはワード固定）。上の例ではd0.w×10，d0.w×d1.wをd0.lに求めている。muluが無符号演算で，mulsが符号付きなのはdivu，divsの関係と同様だ。オーバーフローはありえない（$FFFF_H$を自乗しても32ビットで収まる）ので，演算の結果ccrのCビット，Vビットは常にリセットされる。

乗算を行う専用命令が用意されているのは確かに便利なのだが，68000の乗算命令は内部ではもっと原始的な処理に展開されているらしく，addなどの単純な命令よりもかなり実行時間がかかる。最悪の場合20倍近い。このため乗算命令を使うまでもないような簡単なケースでは加減算やシフトを利用して積を計算することがよくある。表1にd0.wを定数倍するパターンをいくつか挙げてみた。複数の実現法が考えられる場合には最も実行時間が短くなるものを選んである(はず)。ただし，ここでは符号とオーバーフローは考慮していない。また，作業用にd1を使っていることがある。

表1 d0.wを定数倍するパターン

2倍	add.w	d0,d0
3倍	move.w	d0,d1
	add.w	d0,d0
	add.w	d1,d0
4倍	add.w	d0,d0
	add.w	d0,d0
5倍	move.w	d0,d1
	add.w	d0,d0
	add.w	d0,d0
	add.w	d1,d0
6倍	move.w	d0,d1
	add.w	d0,d0
	add.w	d1,d0
	add.w	d0,d0
7倍	move.w	d0,d1
	lsl.w	#3,d0
	sub.w	d1,d0
8倍	lsl.w	#3,d0
16倍	lsl.w	#4,d0
32倍	lsl.w	#5,d0
256倍	lsl.w	#8,d0
1024倍	moveq.l	#10,d1
	lsl.w	d1,d0

▶3月号77ページの藤原さんへ，私の記憶では「北風小僧の寒太郎」は堺正章が歌っていたように思います。
加納 卓也 (21) X1turbo II 徳島県

move.w　d1, −(sp)
clr.w　　−(sp)

とすべきところを，d1.lの上位ワードをあらかじめ0クリアしておくことで，

move.l　d1, −(sp)

の1命令に置き換えている。

169行から先読み入力処理ではバッファが空でなければ読み出し位置から1バイト取り出して返し，空であれば0ではなくEOFを返している。これはTAPの仕様である。また，181行の入力バッファクリア処理はリングバッファならではの単純さだ。リングバッファでは読み出し位置と書き込み位置を一致させるだけで，バッファが空になるわけだ。

TAPDRV改良版

では，最後におまけとして，TAPのバッファサイズをCONFIG.SYSで指定できるよう拡張する方法を示す。リスト3の199行以下をリスト4に置き換えると，起動時のパラメータを解釈してバッファサイズを決める処理が追加される。数値の表示に以前作ったサブルーチンputdecを使っているので忘れずにリンクしてもらいたい。バッファサイズは1Kバイト単位で，

DEVICE = TAPDRV.SYS #/B64

のように指定するようになっている。“#/B”というのが冗長だが，これはPRNDRV.SYSやオリジナル版TAPに合わせたためだ。

プログラム中では，10進文字列から数値へ変換する処理を行う85行以下のサブルーチンatoiと，1Kバイトの単位からバイトに変換するためにビットシフトを利用して1024倍している18〜19行はきちんと理解しておいてもらいたい。両者ともに，コラム「乗算」が参考になるだろう。

＊

というあたりで，デバイスドライバの話はおしまいだ。自信なげに予告してあった“デバイスドライバをあとから組み込む方法”はやはりゆとりがなくてそこまで手が回らなかった。また機会があったら取り上げてみたい。

リスト3 TAPDRV.S

```
  1: *         TAPドライバ(バッファサイズ固定版)
  2:
  3:           .include        doscall.mac
  4:           .include        const.h
  5:           .include        driver.h
  6: *
  7:           .text
  8:           .even
  9: *
 10: *         デバイスヘッダ
 11: *
 12: device_header:
 13:           .dc.l   -1
 14:           .dc.w   CHAR_DEVICE+DISABLE_IOCTRL+COOKED_MODE
 15:           .dc.l   strategy_entry
 16:           .dc.l   interrupt_entry
 17:           .dc.b   'TAP     '
 18: *                  12345678
 19: *
 20: request_header:                         *リクエストヘッダ待避領域
 21:           .dc.l   0
 22:
 23: *
 24: *         ストラテジルーチン
 25: *
 26: strategy_entry:
 27:           move.l  a5,request_header         *リクエストヘッダへのポインタを
 28:                                             *  待避して
 29:           rts                               *速やかに戻る
 30:
 31: *
 32: *         割り込みルーチン
 33: *
 34: interrupt_entry:
 35:           movem.l d0-d2/a0-a1/a4-a5,-(sp) *レジスタ待避
 36:
 37:           movea.l request_header,a5         *a5=リクエストヘッダ
 38:
 39:           moveq.l #0,d0                     *d0.l=コマンドコード
 40:           move.b  CMD_CODE(a5),d0           *
 41:           add.w   d0,d0                     *2倍する
 42:           add.w   d0,d0                     *2倍の2倍で4倍
 43:           lea.l   jump_table,a4             *a4=ジャンプテーブル先頭
 44:           adda.l  d0,a4                     *a4=コマンド処理ルーチンへの
 45:                                             *  ポインタへのポインタ
 46:           movea.l (a4),a4                   *a4=コマンド処理ルーチンへの
 47:                                             *  ポインタ
 48:           jsr     (a4)                      *a4の指すアドレスを
 49:                                             *  サブルーチンコール
 50:
 51:           move.b  d0,ERR_LOW(a5)            *終了ステータスをセット
 52:           lsr.w   #8,d0                     *
 53:           move.b  d0,ERR_HIGH(a5)           *
 54:
 55:           movem.l (sp)+,d0-d2/a0-a1/a4-a5 *レジスタ復帰
 56:           rts                               *Humanへ戻る
 57:
 58: *
 59: *         コマンド処理ジャンプテーブル
 60: *
 61: jump_table:
 62:           .dc.l   init            *0        初期化
 63:           .dc.l   notcmd          *1        (無効)
 64:           .dc.l   notcmd          *2        (無効)
 65:           .dc.l   ioctrl_in       *3        IOCTRLによる入力
 66:           .dc.l   input           *4        入力
 67:           .dc.l   sense           *5        1バイト先読み入力
 68:           .dc.l   inpstat         *6        入力ステータスをチェック
 69:           .dc.l   flush           *7        入力バッファをクリア
 70:           .dc.l   output          *8        出力(VERIFY OFF)
 71:           .dc.l   voutput         *9        出力(VERIFY ON)
 72:           .dc.l   outstat         *10       出力ステータスをチェック
 73:           .dc.l   notcmd          *11       (無効)
 74:           .dc.l   ioctrl_out      *12       IOCTRLによる出力
 75:
 76: *
 77: *         各コマンドの処理
 78: *
 79:
 80: *
 81: *         無効(コマンドコード1,2,11)
 82: *         IOCTRLによる入力(コマンドコード3)
 83: *         IOCTRLによる出力(コマンドコード12)
 84: *
 85: notcmd:
 86: ioctrl_in:
 87: ioctrl_out:
 88:           move.w  #ILLEGAL_CMD,d0           *エラーコードを持って
 89:           rts                               *  戻る
 90:
 91: *
 92: *         入力(コマンドコード4)
 93: *
 94: input:
 95:           move.l  DMA_LEN(a5),d2            *入力要求が0バイトであれば
 96:           beq     done                      *  何もせずに戻る
 97:                                   *そうでなければ
 98:           movea.l readptr,a0                *a0=読み出し位置
 99:           movea.l DMA_ADR(a5),a4            *a4=データ読み込み領域
100:                                             *d2.l=入力要求バイト数
101:
102: inp0:     cmpa.l  writeptr,a0               *データがもうなければ
103:           beq     empty                     *  ループを抜ける
104:           move.b  (a0)+,(a4)+               *1バイト転送
105:           cmpa.l  buffend,a0                *ポインタがバッファ最後を
106:           bcs     inp1                      *  越えたら
107:           lea.l   bufftop,a0                *  先頭を指すように修正する
108: inp1:     subq.l  #1,d2                     *ループカウンタd2.lが0になるまで
109:           bne     inp0                      *  繰り返す
110:
111:           move.l  a0,readptr                *読み出し用ポインタ更新
112:
113: done:     moveq.l #0,d0                     *正常終了
114:           rts
115:
116: empty:    move.b  #EOF,(a4)                 *バッファが空の場合は
117:                                             *  EOFを返す
118:           bra     done                      *正常終了
119:
120: *
121: *         VERIFY OFF時の出力(コマンドコード8)
122: *         VERIFY ON時の出力(コマンドコード9)
123: output:
124: voutput:
```


▶3月号の荒木さんへ。「ザ・すだち」は毒物ではないと思います。ただ，すだちの果汁のタグイをそのまま飲むと毒物になると思いますので，何かで割って飲んでください。
高麗　道也（18）X1turbo　徳島県

```
125:          move.l  DMA_LEN(a5),d2              *入力要求が0バイトであれば
126:          beq     done                        *  何もせずに戻る
127:                                        *そうでなければ
128:          movea.l writeptr,a0                 *a0=次に書き込む位置
129:          movea.l readptr,a1                  *a1=次に読み出す位置
130:          movea.l DMA_ADR(a5),a4              *a4=出力データ
131:                                              *d2.l=出力要求バイト数
132:
133:          moveq.l #0,d1                       *作業用レジスタをクリア
134:
135: out0:    move.b  (a4)+,d1                    *1バイト取り出す
136:          move.b  d1,(a0)+                    *バッファに追加
137:
138:          cmpi.b  #EOF,d1                     *EOFコードは画面クリアの
139:                                              *  コントロールコードなので
140:          beq     out1                        *  表示はしない
141:
142:          move.l  d1,-(sp)                    *1バイト画面に出力
143:          DOS     _CONCTRL                    *
144:          addq.l  #4,sp                       *
145:
146: out1:    cmpa.l  buffend,a0                  *ポインタがバッファ最後を
147:          bcs     out2                        *  越えたら
148:          lea.l   bufftop,a0                  *  先頭を指すように修正する
149:
150: out2:    cmpa.l  a1,a0                       *書き込み位置が読みだし位置に
151:          bne     out3                        *  追いついてしまった場合は
152:          addq.l  #1,a1                       *  読みだし位置を強制的にずらす
153:
154:          cmpa.l  buffend,a1                  *その結果読み出し位置が
155:          bcs     out3                        *  バッファ最後を越えたら
156:          lea.l   bufftop,a1                  *  先頭を指すように修正する
157:
158: out3:    subq.l  #1,d2                       *ループカウンタd2.lが0になるまで
159:          bne     out0                        *  繰り返す
160:
161:          move.l  a0,writeptr                 *書き込み用ポインタ更新
162:          move.l  a1,readptr                  *読み出し用ポインタ更新
163:
164:          bra     done                        *正常終了
165:
166: *
167: *        1バイト先読み入力（コマンドコード5）
168: *
169: sense:
170:          moveq.l #EOF,d0                     *仮にEOFコードを入れておく
171:          movea.l readptr,a0                  *読み出しポインタと
172:          cmpa.l  writeptr,a0                 *  書き込みポインタが
173:          beq     sense0                      *  等しければバッファは空
174:          move.b  (a0),d0                     *そうでなければ何かあるから
175:                                              *  ポインタは固定のまま取り出す
176: sense0:  move.b  d0,SNS_DATA(a5)             *先読みデータをセット
177:          bra     done                        *正常終了
178:
179: *
180: *        入力バッファクリア（コマンドコード7）
181: flush:
182:          move.l  writeptr,readptr            *書き込み位置と
183:                                              *  読み出し位置を一致させる
184:          bra     done                        *正常終了
185:
186: *
187: inpstat:                    *入力ステータスチェック（コマンドコード6）
188: outstat:                    *出力ステータスチェック（コマンドコード10）
189:          bra     done                        *正常終了（常に入出力可）
190: *
191: readptr:
192:          .dc.l   bufftop                     *次に読み出す位置を指すポインタ
193: writeptr:
194:          .dc.l   bufftop                     *次に書き込む位置を指すポインタ
195: buffend:
196:          .dc.l   0                           *バッファ最終アドレス+1
197: *
198: *        ↓以下をバッファとして使用
199: bufftop:
200:
201: BUFFSIZE         =       16*1024             *バッファのバイト数
202:
203: *
204: *        初期化部（コマンドコード0）
205: *
206: init:
207:          pea.l   title                       *タイトルを表示
208:          DOS     _PRINT                      *
209:          addq.l  #4,sp                       *
210:
211:          lea.l   bufftop+BUFFSIZE,a4         *a4 = バッファ最終アドレス+1
212:          move.l  a4,buffend                  *
213:          move.l  a4,DEV_END_ADR(a5)          *デバイスドライバで使用する
214:                                              *  メモリの最終アドレスをセット
215:
216:          bra     done                        *正常終了
217: *
218:          .data
219:          .even
220: *
221: title:                                       *タイトルメッセージ
222:          .dc.b   CR,LF,'TAP DRIVER for X68000',CR,LF
223:          .dc.b   'TAPのデバイス名で入出力が行えます',CR,LF,0
224: *
225:          .end
```

リスト4 TAPDRV.S追加部

```
  1: bufftop:
  2:
  3:          .xref   putdec                      *外部参照
  4: *
  5: *        初期化部（コマンドコード0）
  6: *
  7: init:
  8:
  9:          pea.l   title                       *タイトルを表示
 10:          DOS     _PRINT                      *
 11:          addq.l  #4,sp                       *
 12:
 13:          bsr     getbufsiz                   *バッファサイズ取得
 14:
 15:          move.w  d0,-(sp)                    *(sp)=バッファサイズ（単位K）
 16:
 17:          lea.l   bufftop,a4                  *a4=バッファ先頭
 18:          moveq.l #10,d1                      *1024倍
 19:          lsl.l   d1,d0                       *
 20:                                              *d0.l=バッファサイズ
 21:          add.l   d0,a4                       *a4=バッファ最終+1
 22:          move.l  a4,buffend                  *
 23:          move.l  a4,DEV_END_ADR(a5)          *デバイスドライバで使用する
 24:                                              *  メモリの最終アドレスをセット
 25:
 26:          bsr     putdec                      *バッファサイズを表示
 27:          addq.l  #2,sp                       *
 28:
 29:          pea.l   mes2                        *初期化完了メッセージを表示
 30:          DOS     _PRINT                      *
 31:          addq.l  #4,sp                       *
 32:
 33:          bra     done                        *正常終了
 34:
 35: *
 36: *        CONFIG.SYSで指定されたバッファサイズを取得する
 37: *
 38: getbufsiz:
 39:          movea.l PAR_PTR(a5),a4              *a4=パラメータ先頭
 40:
 41: *        'TAPDRV.SYS',0,'#/B64',0,0
 42: *         ^a4
 43:
 44: skip:    tst.b   (a4)+                       *ファイル名を飛ばす
 45:          bne     skip                        *
 46:
 47: *        'TAPDRV.SYS',0,'#/B64',0,0
 48: *                       ^a4
 49:
 50:          tst.b   (a4)                        *パラメータがなければ
 51:          beq     default                     *  デフォルト値を使う
 52:
 53:          cmpi.b  #'#',(a4)+                  *'#/B'の並びを順にチェック
 54:          bne     inierr                      *
 55:          cmpi.b  #'/',(a4)+                  *
 56:          bne     inierr                      *
 57:          move.b  (a4)+,d0                    *
 58:          andi.b  #%1101_1111,d0              *大文字化
 59:          cmpi.b  #'B',d0                     *
 60:          bne     inierr                      *
 61:
 62: *        'TAPDRV.SYS',0,'#/B64',0,0
 63: *                          ^a4
 64:
 65:          bsr     atoi                        *文字列→数値変換
 66:          tst.w   d0                          *0なら
 67:          beq     inierr                      *  正しくない
 68:          cmpi.l  #1024+1,d0                  *上限のチェック
 69:          bcc     inierr                      *
 70:
 71:          rts                         *d0.w=バッファサイズ（単位K）
 72: *
 73: inierr:
 74:          pea.l   mes1                        *エラーメッセージを表示
 75:          DOS     _PRINT                      *
 76:          addq.l  #4,sp                       *
 77: *        bra     default
 78: default:
 79:          moveq.l #16,d0                      *仮に16Kバイト確保
 80:          rts
 81:
 82: *
 83: *        文字列→数値変換
 84: *
 85: atoi:
 86:          moveq.l #0,d0                       *結果を入れるd0.lをクリア
 87:          moveq.l #0,d1                       *
 88: atoi0:   move.b  (a4)+,d1                    *1文字取り出す
 89:          subi.b  #'0',d1                     *文字→数値変換
 90:          bcs     atoi1                       *
 91:          cmpi.b  #'9'+1,d1                   *
 92:          bcc     atoi1                       *
 93:          mulu.w  #10,d0                      *10倍して
 94:          add.w   d1,d0                       *  1桁追加
 95:          bra     atoi0                       *繰り返す
 96: atoi1:   rts
 97: *
 98:          .data
 99:          .even
100: *
101: title:                                       *タイトルメッセージ
102:          .dc.b   CR,LF,'TAP DRIVER for X68000',CR,LF,0
103: mes1:    .dc.b   'パラメータの指定に誤りがあります',CR,LF
104:          .dc.b   'バッファサイズは以下の形式で指定します',CR,LF
105:          .dc.b   TAB,'DEVICE = TAPDRV.SYS #/Bn',CR,LF
106:          .dc.b   TAB,'（nは1Kバイト単位）',CR,LF
107:          .dc.b   '仮に',0
108: mes2:    .dc.b   'Kバイトのバッファを確保しました',CR,LF
109:          .dc.b   'TAPのデバイス名で入出力が行えます',CR,LF,0
110: *
111:          .end
```

▶ X68000を買うため，休みにはいるたびバイトをするが，なかなかうまくいかない。いいバイトないかなぁ。
形幅　哲也（17）なし　香川県

エレベータのシミュレータ(2)

Izumi Daisuke 泉 大介

今月は「賢いエレベータ作り」の後編。エレベータが2台になっただけではなく，速度も1.5倍ほど速くなりました。さらに待ち時間を折れ線グラフで表示してくれるという至れり尽くせりのサービス。これでイライラも解消されるかも?

先月はコンピュータシミュレーションの例題としてエレベータを作ってみました。各階に到着する乗客をエレベータが黙々と運ぶ様子は，見ているだけで楽しいものです。さて，今月はエレベータの数を増やし，効率的なエレベータの運用を考えてみることにしましょう。

エレベータの数を増やす

ではさっそくエレベータの数を増やす作業にとりかかりましょう。これは実に簡単です。先月号ではエレベータは，

```
int elevator   /* エレベータのいる階
int vector     /* 動いている方向
int pass       /* 乗客
int wait       /* 乗り降り時の待ち
```

という4つの変数で表現されていました。これを複数台分用意すればいいだけです。配列で表現することにしましょう。

```
int Elevator( 1 )   /* エレベータのいる階
int Vector( 1 )     /* 動いている方向
int Pass( 1 )       /* 乗客
int Wait( 1 )       /* 乗り降り時の待ち
```

これで終わりです。あとは，プログラム中で4つの変数を使っているところを適宜書き換えていけばOK。エレベータを表す配列名が大文字になっている理由は，書き換え後のプログラムリストを見ればわかっていただけるでしょう。最後のページに掲載しましたので，どのように書き換えてあるのかちょっとのぞいてみましょう。プログラムリスト3です。

1550行にはエレベータに乗っている人を降ろす関数getOffがあります。今月号のgetOff関数は，乗客を降ろすのはどちらのエレベータなのかを引数として受け取るようになっています。行末に「変更」とコメントしてありますね。次に1580～1610行はプログラムを追加した部分です。先月使ったelevator，vector, passという変数を宣言し，Elevator, Vector, Pass配列から該当エレベータの分をコピーしています。これによって，1620行以降のプログラムは先月のままで，今月の拡張に対応することができるわけです。getOff関数の中で変更を加えられた変数は，1860，1870行で元に戻しています。1850行はWait関係の変更です。小さな変更ですので，ここではわざわざwait変数を宣言せずに対応しています。行末に「変更」とコメントしてありますね。

このように「追加」「変更」のコメントのある場所を順次書き換えていけば，先月号のプログラムリストを入力してある方は簡単に今月号のものを手に入れることができます。関数の並び方がお手元のものとは違うかもしれませんが，それはそのままでOKです。関数ごとに修正を加えていってください。

エレベータがどのように動くのかはsimulation関数を見ればすぐにわかります。3130～3210行のfor～nextループで，エレベータの0番と1番を交互に動かしているだけです。

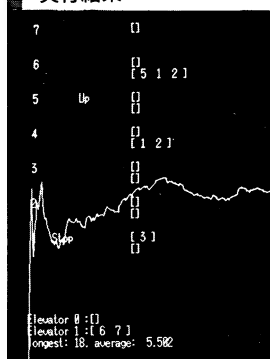
先月は関数ごとにプログラムリストを掲載し，次第にプログラムができあがっていく様子をお見せしました。あとで全部まとめてみると300行以上ものプログラム。皆さんはこれを入力していたわけです。すごいですね。

エレベータのアルゴリズム

プログラムの修正が終わったらさっそくrunしましょう。どうですか，とりあえずこれで動きますね。ではエレベータの動きを追いながら，どのように動かせばいいのかを考えてみることにしましょう。

最初エレベータは止まっています。ある階に人が到着すると，エレベータ0が動き出します。このとき同時にエレベータ1も動き始めるのに気づかれると思います。先月，エレベータを1台だけ動かしたのと同じアルゴリズムを使っているのですから当然です。上の階に向かう人が乗っていたり，上の階で人が待っているとエレベータは上へ動くのです。もう1台のエレベータがなにをやっているのか，ここでは全く考えていません。結果，誰も乗っていないエレベータが（運が悪いと）7階まで動いていくことになります。これは大いに改善する余地がありま

実行結果

▶統合軍愛媛支部，戦艦「羊羹」の艦長M・ドイは3月18日航海日誌にこう書いた。「(で)が月マにやってきた」 土井 政史 (18) X1turbo, MZ-2200 愛媛県

す。こんな無駄なことはないでしょう。

これに関して注意してほしいのは，プログラムリスト3の2350～2420行です。先月のプログラムでは，エレベータが7階に到着した場合には必ず7階に向かう人がエレベータに乗っているか，あるいは7階で人が待っていました。どちらにしてもエレベータは7階でいったん停止しますから，階を越えてエレベータが動き続けようとすることなどなかったのです。しかし，プログラムリスト3ではこのような事態が発生してしまいます。そこでこれらの追加によってエレベータが屋上に出てしまわないよう，地下にもぐってしまわないよう，修正しているのです。

プログラムリスト1はこの無駄を排除するために作った「方向決定関数」です。direction1と命名しました。試行錯誤の結果，direction関数に簡単な追加を施すことでうまく解決できました。先月と同じように，適当な行番号から（たとえば4000行から）4001，4002とひとつずつ行番号を増やしながら入力すると見比べやすく簡単でいいでしょう。頭に仮に付けてある行番号は省いて入力してください。

では無駄排除の方法を説明していきましょう。3行ではdownward, upwardの2つの変数が宣言してあります。これはもう一方のエレベータが上に向かっているか，下に向かっているかを示すフラグです。エレベータAより上の階にエレベータBがいて，しかもエレベータBが上に向かって動いているときにupwardフラグが1になります。downwardフラグはこの逆です。これらのフラグをセットしているのが7～15行のfor～nextループです。続く17行からのif文は，エレベータAが止まっているとき，つまりvectorが0のときにはその階に人がいるかどうか，上か下の階に待っている人がいるかどうかを調べて動く方向を決定します。これはdirection関数と同じです。次にエレベータAが止まっている階に人がいない場合の方向決定ですが，ここに少々細工をしてやります。

upwardが1のとき，つまりエレベータBがエレベータAより上の階にいて上に動いているときには，エレベータAは「下に待っている人がいるかどうか」を調べます。もしエレベータBが下の階にいて下に動いているときには「上の階で人が待っているかどうか」だけを調べます（22～25行）。他方のエレベータが上に動いているなら，エレベータAより上の階で待っていて上に向かう人はエレベータBに乗ってしまったあとである可能性が高く，またエレベータAより上の階で下に向かう人はエレベータBが7階まで行ったあと降りてきて拾う可能性が高いと判断したからです。

次の26～30行では，downwardが0のとき，つまりエレベータBが下にいて下に向かっていないときと，upwardが0のときのエレベータAの動きを決めています。downwardが0のときというのは，エレベータBがエレベータAより下の階にいて上に向かっているか，エレベータBがエレベータAより上の階にいる場合です。このとき下の階で待っている人がいればエレベータAは下に迎えに行きます。upwardが0の場合にはエレベータAは上に迎えに行きます。

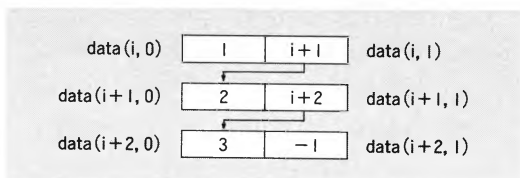
これらの変更・追加の結果，エレベータは次のような動作をするようになります。最初2台のエレベータは1階に止まっています。上の階に乗客が到着すると1台のエレベータが動きだします。upwardが1になりますからもう1台は17～30行のどの条件にもマッチせず止まったままになります。動いているエレベータが乗客を乗せたり降ろしたりしながら7階に到着し，下向きに動きだしたときもう1台のエレベータは28行のif文が成立して動き始めます。あるいは，1階に乗客が到着した場合にももう1台は動き始めます。7階で待っているひとりの人を乗せるため2台のエレベータが動き出すという間抜けな事態がこれで回避されるという寸法です。

では新しいdirection関数を試してみることにしましょう。プログラムリスト3に続いて，4000行か

プログラムリスト1
ちょっと賢いdirection関数

```
 1: func direction1( elvNo )
 2:   int elevator, vector, pass
 3:   int downward = 0, upward = 0  /* 追加
 4:   elevator = Elevator( elvNo )
 5:   vector = Vector( elvNo )
 6:   pass = Pass( elvNo )
 7:   for i=0 to 1                  /* for文追加
 8:     if i = elvNo then continue
 9:     if Elevator( i ) < elevator and Vector( i ) = -1 then {
10:       downward = 1
11:     }
12:     if Elevator( i ) > elevator and Vector( i ) = 1 then {
13:       upward = 1
14:     }
15:   next
16:   /*
17:   if vector = 0 then {
18:     if floor( elevator, 0 ) <> EMPTY then {
19:       vector = -1
20:     } else if floor( elevator, 1 ) <> EMPTY then {
21:       vector = 1
22:     } else if upward and lower( elevator ) then {   /* 追加
23:       vector = -1
24:     } else if downward and upper( elevator ) then { /* 追加
25:       vector = 1
26:     } else if downward=0 and lower( elevator ) then { /* 変更
27:       vector = -1
28:     } else if upward=0 and upper( elevator ) then {   /* 変更
29:       vector = 1
30:     }
31:   } else {
32:     if pass = EMPTY then {
33:         if vector = 1 then {
34:           if floor( elevator, 0 ) <> EMPTY then {
35:             if floor( elevator, 1 ) = EMPTY then {
36:               if upper( elevator ) = 0 then vector = -1
37:             }
38:           }
39:         } else if vector = -1 then {
40:           if floor( elevator, 1 ) <> EMPTY then {
41:             if floor( elevator, 0 ) = EMPTY then {
42:               if lower( elevator ) = 0 then vector = 1
43:             }
44:           }
45:         }
46:     }
47:   }
48:   Vector( elvNo ) = vector
49: endfunc
```

▶わ…私は感動です。いままでいろんな雑誌の応募用ハガキを見てきましたが，Oh！Xのハガキだけです。まともに学校名（愛媛大学農学部付属農業高等学校）を書けるスペースがあったのは……（なんたってフルネームで15字）。　　　住友　智代(17) X1 愛媛県

図1　リストの内部表現

ら4001，4002と行番号を増やしながら4049行まで入力します。入力が終わったらsimulation関数に変更を加えます。プログラムリスト3の，

```
3150        direction( i )
```

を，

```
3150        direction1( i )
```

に変更してください。

実行結果はどうですか？　かなり無駄が省かれたことと思います。もちろんまだ満足のいくものではありませんが，direction関数と比べると随分洗練された動きをしています。さて，乗客の待ち時間のほうはどうなっているのでしょうか。新しいdirectin1関数によって短くなったのでしょうか。次にこれを調べることにしましょう。

待ち時間を計る

先月，乗客を保持するために新しくリストというデータ構造を導入しました。それが実際にどのような形で実現されているのかはブラックボックスとして，リストを操作することだけに主眼をおいて説明しました。ここでリストというデータ構造がどのようにして内部で実現されているのかを説明しておきたいと思います。このデータ構造に少し変更を加えるだけで簡単に待ち時間を計ることができるからです。

リストは伸縮自在な配列のようなものです。簡単にするため，整数しか保持できないようにしたことは先月説明したとおりです。このリスト構造はdataという配列によって実現されています。実現方法は図1です。data配列は2次元の配列です。横方向に2つの箱が並んだものが1000個分用意されています。data配列の左側の箱data (i, 0) には乗客を入れます。そして右側の箱data (i, 1) には「次のデータを入れてある配列の添字」が入っています。データの続きを表しているわけです。図1は [1 2 3] というリストを意味しています。最後の部分であるdata (i+2, 1) には−1，つまりEMPTYを入れ，続くデータがないという印にします。head, tailの2つの関数がなにをやっているのか調べてみてください。

問題の待ち時間ですが，横に2つの箱が並んでいるdata配列を変更し，箱の数を3つにすれば解決できます。最後の箱には乗客がエレベータを待ち始めた時刻を入れておきます。エレベータに乗ったときにそのときの時刻から待ち始めの時刻を引いてやれば，どれだけの時間エレベータを待っていたのかがわかるという仕組みです。時刻ですが，これはtimerという変数を用意し，これを増やしていくことで表現することにしましょう。

プログラムリスト2を見てください。Aは大域変数の変更と追加です。data配列の大きさを変更し，3〜7行の変数を追加します。passesはエレベータに乗った乗客の総数，longestは一番長く待った人の待ち時間，averageは待ち時間の平均値を入れる変数です。lastXとlastYは待ち時間の平均値をグラフで表示するために用意しました。前回グラフを描いた座標を入れます。Aの1行はプログラムリスト3の10行の変更です。3〜7行は，プログラムリスト3の113〜117行部分に入力してください。大域変数はプログラム先頭になければならないからです。

プログラムリスト2のBは関数の変更と，待ち時間計測用の新しい関数を追加しています。init関数はグラフィック画面の初期化を追加しました。arrival関数は乗客が到着したときに，setTime関数を呼び出して現在の時刻をセットする部分の追加です。getOn関数では乗客を乗せるときに，待ち時間の平均を計算するためgetAverage関数を呼び出すようにしてあります。

simulation関数ではwhile〜endwhileのループを1回まわるごとにtimerをひとつ増やすように変更を加えました。また無限ループをtimerが768より小さい間だけループするように書き換え，グラフが画面右端まで表示されたら実行を終了するようにしてあります。86行をコメントにしたのはグラフをさっさと描かせたかったからです。これまでのように1秒のウエイトを入れたければコメントをはずしてく

リストにおけるデータ削除

[1 2 3] というリストから2を削除する場合，

```
data (i, 1)=i+2
```

とすればよい。本文の図1を見ながら考えてみていただきたい。注意しなければならないのは，このときdata (i+1, n) は誰からもつながらない宙ぶらりんデータとなってしまうことだ。先月摩訶不思議な作用でゴミが発生するといったが，これがゴミの正体である。このような方法でデータ削除を続けると宙ぶらりんデータが次第に増えていき，最後にはdata配列すべてがゴミになってしまうことになる。

これを避けるため本プログラムでは次のような方法を採用している。initList関数はプログラム起動時にdata配列をひとつの長いリストにする関数である。この長いリストを自由リストといい，その先頭は変数dataBaseに保持されている。addData関数によって新しいリストを作る必要が生じた場合，addData関数はnewData関数を呼び出して自由リストの先頭からデータを入れる箱をひとつ切り出してもらう。結果自由リストはひとつ減ることになり，dataBaseの値も更新される。

逆にリストから要素を削除する場合には，delData関数はgc関数を呼び出して削除した箱を自由リストに戻すという作業を行う。この結果自由リストはひとつ増えることになる。要素の削除には必ずdelData関数を使うように先月指示したのは，このゴミ回収作業を行うためである。

newData，gcの2つの関数の働きにより，シミュレーション実行中に1000人を超える人がエレベータに殺到しない限り，自由リストが枯渇することはない。

ださい。

これらの変更は小さいので，「3205行」のように中途半端な行番号を使うことで簡単に修正できるでしょう。

94行からは時間計測用の新しい関数です。4000行以降にはdirection1関数を入力してありますので，これらは5094行から1行ずつ入力していくといいでしょう。setTime関数は，引数として受け取ったリストの先頭要素に現在の時刻を付け加えます。逆にgetTime関数は引数として受け取ったリストの先頭要素から時刻を取り出します。試してみましょう。プログラムリスト2を先の指示にしたがって入力したらrunしてください。「キーを押してください」と表示されたらBREAKキーで実行を中断します。ここで，

```
timer = 300
x=addData (1,−1) /* xに [1] をセット
setTime ( x )
```

とすると，リストxの先頭要素に現在の時刻300がセットされます。これは1階に向かう人が時刻300にやってきたことを意味します。

```
timer = 350
x=addData (2,x)  /* x= [2 1]
setTime ( x )
```

なら2階に向かう人に時刻350がセットされます。確かめてみましょう。

```
print getTime( x )          /* 2階に行く人
```

なら350が表示されますし，

```
print getTime(tail( x ))  /* 1階に行く人
```

なら300が表示されます。

最後にgetAverage関数です。この関数はリストを引数に受け取り，先頭要素から順にエレベータ待ちを始めた時刻を取り出しながら，平均待ち時間と最長待ち時間の計算を行います。104～110行で計算が終わったら，112行で画面に表示し113行でグラフを描いています。

プログラムリスト2
待ち時間を計る

```
A） 大域変数の変更と追加

  1: int data( 1000, 2 ) /* 変更
  2: /*
  3: int   timer = 0     /* 追加
  4: int   passes = 0    /* 追加
  5: int   longest = 0   /* 追加
  6: float average = 0   /* 追加
  7: int   lastX, lastY  /* 追加

B） 関数の変更と新しい関数の追加

  1: func init()
  2:   int i
  3:   for i=1 to 7
  4:     floor( i, 0 ) = EMPTY
  5:     floor( i, 1 ) = EMPTY
  6:   next
  7:   /*
  8:   print "キーをおしてください"
  9:   while inkey$(0)=""
 10:     rnd()
 11:   endwhile
 12:   screen 2,0,1,1      /* 追加
 13:   lastX = 0           /* 追加
 14:   lastY = 511         /* 追加
 15:   display()
 16:   moveElevator( 0 )
 17:   moveElevator( 1 )
 18: endfunc
 19: /*
 20: func arrival()
 21:   int ff /* floorFrom
 22:   int ft /* floorTo
 23:   if rnd() > 0.3 then return()
 24:   /*
 25:   ff = int( rnd()*7 ) + 1
 26:   repeat
 27:     ft = int( rnd()*7 ) + 1
 28:   until ff <> ft
 29:   if ff < ft then {
 30:     floor( ff, 1 ) = addData( ft, floor( ff, 1 ))
 31:     setTime( floor( ff, 1 ))     /* 追加
 32:     locate locArvl, (7-ff)*3
 33:     print chr$(5);
 34:     printList( floor( ff, 1 ))
 35:   } else {
 36:     floor( ff, 0 ) = addData( ft, floor( ff, 0 ))
 37:     setTime( floor( ff, 0 ))     /* 追加
 38:     locate locArvl, (7-ff)*3+1
 39:     print chr$(5);
 40:     printList( floor( ff, 0 ))
 41:   }
 42: endfunc
 43: /*
 44: func getOn( elvNo )
 45:   int flag = 0
 46:   int elevator, vector, pass
 47:   elevator = Elevator( elvNo )
 48:   vector = Vector( elvNo )
 49:   pass = Pass( elvNo )
 50:   if Wait( elvNo ) = 1 then return()
 51:   if vector = 1 and floor( elevator, 1 ) <> EMPTY then {
 52:     getAverage( floor( elevator, 1 ))    /* 追加
 53:     pass = concat( floor( elevator, 1 ), pass )
 54:     floor( elevator, 1 ) = EMPTY
 55:     locate locArvl, (7-elevator)*3
 56:     flag = 1
 57:   }
 58:   if vector = -1 and floor( elevator, 0 ) <> EMPTY then {
 59:     getAverage( floor( elevator, 0 ))    /* 追加
 60:     pass = concat( floor( elevator, 0 ), pass )
 61:     floor( elevator, 0 ) = EMPTY
 62:     locate locArvl, (7-elevator)*3+1
 63:     flag = 1
 64:   }
 65:   if ( flag ) then {
 66:     printList( EMPTY )
 67:     print chr$(5);
 68:     Wait( elvNo ) = Wait( elvNo ) + 1
 69:   }
 70:   Pass( elvNo ) = pass
 71: endfunc
 72: /*
 73: func simulation()
 74:   str tm
 75:   int i
 76:   while timer < 768   /* 変更
 77:     for i=0 to 1
 78:       arrival()
 79:       direction1( i )
 80:       getOff( i )
 81:       getOn( i )
 82:       moveElevator( i )
 83:       locate 0,24
 84:       tm = time$
 85:     next
 86:     /* while tm = time$ : endwhile /* 変更
 87:     timer = timer+1 /* 追加
 88:   endwhile
 89:   locate 0, 28 /* 追加
 90: endfunc
 91: /*
 92: /* 待時間計測用の関数追加
 93: /*
 94: func setTime( lst )
 95:   data( lst, 2 ) = timer
 96: endfunc
 97: /*
 98: func int getTime( lst )
 99:   return( data( lst, 2 ))
100: endfunc
101: /*
102: func getAverage( lst )
103:   int waitTime
104:   while lst <> EMPTY
105:     waitTime = timer - getTime( lst )
106:     if waitTime > longest then longest = waitTime
107:     passes = passes+1
108:     average = ( waitTime - average ) / passes + average
109:     lst = tail( lst )
110:   endwhile
111:   locate 0, 27
112:   print using "longest:###. average:###.###";longest,average
113:   line( lastX, lastY, timer, 511-average*50, 11 )
114:   lastX = timer
115:   lastY = 511-average * 50
116: endfunc
```

▶貴誌を愛読して6年。今になって気がついた。若者の雑誌であることを。
井門　清 (40) X68000ACE　愛媛県

より賢いエレベータにするには

実行してみた結果はいかがでしたか。最初は待ち時間平均の振動が大きいのですが，時間がたつにしたがって収束してくる様子がわかると思います。２つのdirection関数の性能を比較したければ次のようにするといいでしょう。まず，direction関数を呼び出すようにsimulation関数を変更してグラフを描かせます。次にdirection1関数を呼び出すようにsimulation関数を変更し，init関数のscreen命令をコメントにして実行するのです。screen命令をコメントにするとグラフィック画面がクリアされませんので，２つのdirection関数をグラフで比較できます。グラフは画面左下を原点として表示されますので，下に描かれるグラフのほうが待ち時間が短いことを意味します。direction1関数の性能はどうでしょうか。乱数の発生条件を同じにするため，「キーを押してください」と表示される前に１発キーを叩いておけば万全です。

direction1関数はまだまだ改良の余地があります。エレベータAが４階にいて上に向かっており，エレベータBは１階で止まっているとします。２階あるいは３階に上に向かう人がやって来た場合，エレベータBはすぐに動き始めるべきです。現在はエレベータAが止まるか，１階に上に行く人がこない限りエレベータBは動きません。この改良だけで待ち時間平均はもう少し短くなるはずです。自分と相手のエレベータの間の階で人が待っているかどうかを調べる関数が必要になりますが，これは簡単でしょう。

あるいはもっと徹底的に効率化を図ることもできるはずです。このシミュレーションでは現在エレベータが何階にいるのかを，待っている人が知ることができるタイプのものを想定しましたが，これを待っている人から隠せば好き放題のことができるのです。４階から上に行こうとしている人を無視して，５階から下に行こうとしている人を迎えにエレベータを動かすこともできます。４階で待っている人は自分が無視されたことを知る方法はないのですから，そのほうが効率がいいと判断したら実行することになんの障害もないのです。実際シミュレーション画面を見ていると，こいつら全部無視でここにダイレクトにエレベータを動かせば効率がいいなぁと思うことがしばしばあります。

ひとつ徹底的に効率化を図ったdirection関数を作ってみたのですが，これがもう，嵐のようなif文の塊になってしまいました。まるでエキスパートシステムです。ちょっと大きいので発表は止めにしますが，皆さんも自分なりの効率のいいdirection関数を考え，試してみてください。いいものができたら，編集部気付で私宛に送ってくださいね。できのいい作品はこの連載の中で紹介したいと思います。

来月はちょっと息抜き。ゲームに挑戦してみましょう。

プログラムリスト３
２基のエレベータを動かす

```
 10 int data( 1000, 1 )
 20 int dataBase = 0
 30 int EMPTY = -1
 40 /*
 50 int floor( 7, 1 )
 60 int locArvl = 20
 70 /*
 80 int Elevator( 1 ) = { 1, 1 } /* 変更
 90 int Pass( 1 ) = { -1, -1 }   /* 変更
100 int Vector( 1 ) = { 0, 0 }   /* 変更
110 int Wait( 1 ) = { 0, 0 }     /* 変更
120 /*
130 initList()
140 init()
150 simulation()
160 end
170 /*
180 /*
190 func initList()
200   int i
210   for i=0 to 999
220     data( i, 1 ) = i+1
230   next
240   data( 1000, 1 ) = EMPTY
250 endfunc
260 /*
270 func int newData()
280   int retData
290   retData = dataBase
300   dataBase = tail( dataBase )
310   data( retData, 1 ) = EMPTY
320   return( retData )
330 endfunc
340 /*
350 func int head( lst )
360   return( data( lst, 0 ))
370 endfunc
380 /*
390 func int tail( lst )
400   return( data( lst, 1 ))
410 endfunc
420 /*
430 func int addData( x, lst )
440   int nd
450   nd = newData()
460   data( nd, 0 ) = x
470   data( nd, 1 ) = lst
480   return( nd )
490 endfunc
500 /*
510 func int concat( lst1, lst2 )
520   int lst
530   lst = lst1
540   while tail( lst ) <> EMPTY
550     lst = tail( lst )
560   endwhile
570   data( lst, 1 ) = lst2
580   return( lst1 )
590 endfunc
600 /*
610 func int delData( x, lst )
620   int tmp
630   int retData
640   int i
650   if x = lst then {
660     retData = tail( lst )
670     gc( lst )
680   } else {
690     retData = lst
700     while tail( lst ) <> x
710       lst = tail( lst )
720     endwhile
730     tmp = tail( lst )
740     data( lst, 1 ) = tail( tmp )
750     gc( tmp )
760   }
770   return( retData )
780 endfunc
790 /*
800 func gc( gabbage )
810   data( gabbage, 1 ) = dataBase
820   dataBase = gabbage
830 endfunc
840 /*
850 func printList( lst )
860   print "[";
870   if lst <> EMPTY then {
880     repeat
890       print head( lst );
900       lst = tail( lst )
910     until lst = EMPTY
920   }
930   print "]";
940 endfunc
950 /*
960 func init()
970   int i
980   for i=1 to 7
```


▶THE SOFTOUCHのコーナーはもう少しページ数を多くして〜。
井上　哲郎(24) X68000PRO 高知県

```
 990     floor( i, 0 ) = EMPTY
1000     floor( i, 1 ) = EMPTY
1010   next
1020   /*
1030   print "キーを押してください"
1040   while inkey$(0)=""
1050     rnd()
1060   endwhile
1070   display()
1080   moveElevator( 0 )  /* 変更
1090   moveElevator( 1 )  /* 追加
1100 endfunc
1110 /*
1120 /* Arrival of Passenger
1130 /*
1140 func arrival()
1150   int ff /* floorFrom
1160   int ft /* floorTo
1170   if rnd() > 0.3# then return()
1180   /*
1190   ff = int( rnd()*7 ) + 1
1200   repeat
1210     ft = int( rnd()*7 ) + 1
1220   until ff <> ft
1230   if ff < ft then {
1240     floor( ff, 1 ) = addData( ft, floor( ff, 1 ))
1250     locate locArvl, (7-ff)*3
1260     print chr$(5);
1270     printList( floor( ff, 1 ))
1280   } else {
1290     floor( ff, 0 ) = addData( ft, floor( ff, 0 ))
1300     locate locArvl, (7-ff)*3+1
1310     print chr$(5);
1320     printList( floor( ff, 0 ))
1330   }
1340 endfunc
1350 /*
1360 /* Display Floor
1370 /*
1380 func display()
1390   int i, j
1400   print chr$(12);
1410   for i=1 to 7
1420     j = 8 - i
1430     print j;
1440     locate locArvl, csrlin
1450     printList( floor( j, 1 ))
1460     print
1470     locate locArvl, csrlin
1480     printList( floor( j, 0 ))
1490     print : print
1500   next
1510 endfunc
1520 /*
1530 /* Passengers get off
1540 /*
1550 func getOff( elvNo )             /* 変更
1560   int chk, tmp
1570   int flag = 0
1580   int elevator, vector, pass   /* 追加
1590   elevator = Elevator( elvNo ) /* 追加
1600   vector = Vector( elvNo )     /* 追加
1610   pass = Pass( elvNo )         /* 追加
1620   if pass = EMPTY then return()
1630   chk = pass
1640   while chk <> EMPTY
1650     if head( chk ) <> elevator then {
1660       chk = tail( chk )
1670     } else {
1680       tmp = tail( chk )
1690       pass = delData( chk, pass )
1700       chk = tmp
1710       flag = 1
1720     }
1730   endwhile
1740   if pass = EMPTY then {
1750     if vector = 1 then {
1760       if upper( elevator ) = 0 then {
1770               if floor( elevator, 1 ) = EMPTY then vector = 0
1780       }
1790     } else {
1800       if lower( elevator ) = 0 then {
1810         if floor( elevator, 0 ) = EMPTY then vector = 0
1820       }
1830     }
1840   }
1850   Wait( elvNo ) = Wait( elvNo ) + flag /* 変更
1860   Vector( elvNo ) = vector             /* 追加
1870   Pass( elvNo ) = pass                 /* 追加
1880 endfunc
1890 /*
1900 /* Passengers get on
1910 /*
1920 func getOn( elvNo )              /* 変更
1930   int flag = 0
1940   int elevator, vector, pass   /* 追加
1950   elevator = Elevator( elvNo ) /* 追加
1960   vector = Vector( elvNo )     /* 追加
1970   pass = Pass( elvNo )         /* 追加
1980   if Wait( elvNo ) = 1 then return()  /* 変更
1990   if vector = 1 and floor( elevator, 1 ) <> EMPTY then {
2000     pass = concat( floor( elevator, 1 ), pass )
2010     floor( elevator, 1 ) = EMPTY
2020     locate locArvl, (7-elevator)*3
2030     flag = 1
2040   }
2050   if vector = -1 and floor( elevator, 0 ) <> EMPTY then {
2060     pass = concat( floor( elevator, 0 ), pass )
2070     floor( elevator, 0 ) = EMPTY
2080     locate locArvl, (7-elevator)*3+1
2090     flag = 1
2100   }
2110   if ( flag ) then {
2120     printList( EMPTY )
2130     print chr$(5);
2140     Wait( elvNo ) = Wait( elvNo ) + 1 /* 変更
2150   }
2160   Pass( elvNo ) = pass              /* 追加
2170 endfunc
2180 /*
2190 /* Move Elevator
2200 /*
2210 func moveElevator( elvNo )     /* 変更
2220   int elevator, vector, wait   /* 追加
2230   elevator = Elevator( elvNo ) /* 追加
2240   vector = Vector( elvNo )     /* 追加
2250   wait = Wait( elvNo )         /* 追加
2260   /*
2270   locate (elvNo+1)*5, (7-elevator)*3 /* 変更
2280   print "    "
2290   /*
2300   if wait = 0 then {
2310     elevator = elevator + vector
2320   } else {
2330     wait = 0
2340   }
2350   if elevator < 1 then {        /* 追加
2360     elevator = 1                /* 追加
2370     vector = 0                  /* 追加
2380   }                             /* 追加
2390   if elevator > 7 then {        /* 追加
2400     elevator = 7                /* 追加
2410     vector = 0                  /* 追加
2420   }                             /* 追加
2430   /*
2440   locate 0, 25+elvNo                      /* 変更
2450   print "Elevator";elvNo;":";chr$(5); /* 変更
2460   printList( Pass( elvNo ))               /* 変更
2470   locate (elvNo+1)*5, (7-elevator)*3  /* 変更
2480   print mid$( "DownStopUp  ", 4*(vector+1)+1, 4 )
2490   Elevator( elvNo ) = elevator /* 追加
2500   Vector( elvNo ) = vector     /* 追加
2510   Wait( elvNo ) = wait         /* 追加
2520 endfunc
2530 /*
2540 func direction( elvNo )        /* 変更
2550   int elevator, vector, pass   /* 追加
2560   elevator = Elevator( elvNo ) /* 追加
2570   vector = Vector( elvNo )     /* 追加
2580   pass = Pass( elvNo )         /* 追加
2590   if vector = 0 then {
2600     if floor( elevator, 0 ) <> EMPTY then {
2610       vector = -1
2620     } else if floor( elevator, 1 ) <> EMPTY then {
2630       vector = 1
2640     } else if lower( elevator ) then {
2650       vector = -1
2660     } else if upper( elevator ) then {
2670       vector = 1
2680     }
2690   } else {
2700     if pass = EMPTY then {
2710         if vector = 1 then {
2720           if floor( elevator, 0 ) <> EMPTY then {
2730             if floor( elevator, 1 ) = EMPTY then {
2740               if upper( elevator ) = 0 then vector = -1
2750             }
2760           }
2770         } else if vector = -1 then {
2780           if floor( elevator, 1 ) <> EMPTY then {
2790             if floor( elevator, 0 ) = EMPTY then {
2800               if lower( elevator ) = 0 then vector = 1
2810             }
2820           }
2830         }
2840     }
2850   }
2860   Vector( elvNo ) = vector /* 追加
2870 endfunc
2880 /*
2890 func int upper( n )
2900   int i
2910   for i=n+1 to 7
2920     if floor( i, 0 ) <> EMPTY then return( 1 )
2930     if floor( i, 1 ) <> EMPTY then return( 1 )
2940   next
2950   return( 0 )
2960 endfunc
2970 /*
2980 func int lower( n )
2990   int i
3000   for i=1 to n-1
3010     if floor( i, 0 ) <> EMPTY then return( 1 )
3020     if floor( i, 1 ) <> EMPTY then return( 1 )
3030   next
3040   return( 0 )
3050 endfunc
3060 /*
3070 /* シミュレーション
3080 /*
3090 func simulation()
3100   str tm
3110   int i  /* 追加
3120   while 1
3130     for i=0 to 1 /* 追加
3140       arrival()
3150       direction( i )    /* 変更
3160       getOff( i )       /* 変更
3170       getOn( i )        /* 変更
3180       moveElevator( i ) /* 変更
3190       locate 0,24
3200       tm = time$
3210     next
3220     while tm = time$ : endwhile
3230   endwhile
3240 endfunc
```

▶うちの大学にはFuzzyの山川ちゃんがいます。授業を受けたことはなかったけど「I-MY」を見て今度受けてみようかなぁ，と思いました。
関口　達也（21）X68000EXPERT-HD　福岡県

BACK ISSUES バックナンバー案内

ここには1989年5月号から1990年4月号までをご紹介しました。現在1989年6～12, 1990年1～4月号までの在庫がございます。バックナンバーおよび定期講読のお申し込み方法については，180ページを参照してください。

1989

5月号（品切れ）

特集　MIDIサウンドデータ料理術

LA音源をFM音源でシミュレート/X-BASICでMIDI制御

特別企画　第4回「言わせてくれなくちゃだワ」

- シャープパソコンフォーラム'89 in赤坂
- 詳解Human68k ver.2.0
- MZ-2500，X1/X1turbo用　戦略的ライトサイクルゲーム

連載　C調言語講座PRO-68K/　OS-9/X68000入門　X68000マシン語プログラミング

全機種共通システム　ソースジェネレータRING

6月号

特集　これからのXfamily

X68000に光磁気ディスクを/学習リモコンの製作

THE SOFTOUCH　ライトニングバッカス/Might and MagicII他

- OPMA用外部関数による　KENBAN.BAS
- X1/X1turbo用ドライブゲーム　Spirit of Rally
- X1turboZ用　これ，パズルなんですか。

MZ-2500 MIDI入門(1)MIDIボードを作る

C調言語講座PRO-68K/X68000マシン語プログラミング

全機種共通システム　超小型コンパイラTTC

7月号

特集　3Dグラフィックへの飛翔

Zバッファアルゴリズム/スムースシェイディング 他

THE SOFTOUCH Terazzo PRO-68K/アドヴァンスト・ファンタジアン

新連載
DōGA・CGアニメーション講座
MZ-2500用グラフィックエディタ作成講座
マシン語カクテル in Z80's Bar
X-BASICプログラミング調理実習

全機種共通システム TTC用パズルゲームTIC BAN

X68000マシン語プログラミング/C調言語講座PRO-68K 他

8月号

特集1　X1プログラミングガイドブック

PCGの基礎から奥義まで/超高速ラインルーチン 他

特集2　3Dグラフィックの深淵へ

スキャンラインZバッファ/3Dモデリング 他

新連載（で）のショートプロぱーてぃ

X68000マシン語プログラミング/C調言語講座 PRO-68K

X-BASICプログラミング調理実習/DōGA・CGA講座

MZ-2500用グラフィックエディタ/Z80's Bar 他

全機種共通システム CP/M用ファイルコンバータ

9月号

特集　活用ハードディスク&プリンタ

各社ハードディスク接続総チェック/ハードディスク雑学講座/COPYキーメニュー/ビデオプリンタ活用プログラム 他

THE SOFTOUCH　ジェノサイド/琉球/mFORTH Compiler

- サイバースティックで遊ぶ　**不思議な環境ソフトの世界**
- X1/X1turbo用シューティングゲーム　**Defeat X**

Z80's Bar/MZ-2500グラフィックエディタ 他

[X68000] X-BASIC/マシン語/C調言語講座/DōGA・CGA

全機種共通システム　生物進化シミュレーションBUGS

10月号

特集　ゲーム面白心理学

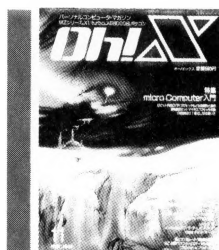
ソーサリアン・宇宙からの訪問者/ファンタジーゾーン
ねじ式/ガウディ・バルセロナの風/サバッシュ 他

- MZ-700用シューティングゲームSide Roll-F
- X1/X1turdo用カードゲームBonding

ショートプロ/Z80's Bar/MZ-2500グラフィックエディタ

X68000マシン語/X-BASIC/C調言語講座/DōGA・CGA

THE SOFTOUCH　Z'sTRIPHONY DIGITAL CRAFT/James68K

全機種共通システム　小型インタプリタ言語TTI

11月号

特集　microComputer入門

初歩からのCPU物語/RISCプロセッサの設計と製作
X68000&X1で周辺LSIを使いこなそう

連載
ショートプロ/Z80's Bar/MZ-2500グラフィックエディタ
X68000マシン語/X-BASIC/C調言語講座/DōGA・CGA

- X68000用カードゲームばばぬき

LIVE in '89　メタルホーク/オブ・ラ・ディ，オブ・ラ・ダ

THE SOFTOUCH　Stationery PRO-68K/リングマスター1

全機種共通システム　TTI用パズルゲームPUSH BON!

12月号

特集　Cプログラミングへの招待

付録　C言語簡易リファレンス

連載
ショートプロぱーてぃ/Z80's Bar
X68000マシン語/X-BASIC/DōGA・CGA

- Oh!X2周年特別企画「素粒子の声が聞こえる」
- X1/turbo用アクションゲームACTIVE UNIT

LIVE in '89　天空の城ラピュタ/ギャラクシーフォース

THE SOFTOUCH　38万キロの虚空/た～みのる2

全機種共通システム　SLANG用リダイレクションライブラリ

1990

1月号

特集1　オペレーティングスタイルの研究

特集2　Cプログラミング応用編

連載
ショートプロぱーてぃ/Z80's Bar
X68000マシン語/C調言語講座/DōGA・CGA

- X1/turbo用シミュレーションゲームSuper Battle

LIVE in '90　さよならを過ぎて/RYDEEN

THE SOFTOUCH　レナム/メタルサイト

全機種共通システム　WORM KUN/再掲載SLANG

特別付録　X68000 THE SOFTWARE CATALOGUE

2月号

特集　画像圧縮へのアプローチ

連載
ショートプロぱーてぃ/Z80's Bar/DōGA・CGA
X68000マシン語/C調言語講座/X-BASIC調理実習

- X68000用ゲームプログラムGon Gon
- MZ-700用紙芝居Eyelarth

LIVE in '90　オーダイン/魔女の宅急便

THE SOFTOUCH　A-JAX/フラッピー2/夢幻戦士ヴァリスII
マジックパレット/Mu-1/CYBERNOTE PRO-68K

全機種共通システム　超小型コンパイラTTC++

3月号

特集　MUSICアドベンチャー

X68000用MIDIドライバ&音源エディタ
なんでも鳴らせるOPMD.X/MMLを楽譜データに

連載
ショートプロぱーてぃ/Z80's Bar/DōGA・CGA
C調言語講座/X-BASIC調理実習

- X1/turboシミュレーションCRISIS in Tokyo

LIVE in '90　パワードリフト/スキーム/となりのトトロ

THE SOFTOUCH　ナイトアームズ/斬/ダンジョンマスター

全機種共通システム　超多機能アセンブラOHM-Z80

4月号

特集　ゲームシステム文学誌

連載
ショートプロぱーてぃ/Z80's Bar/DōGA・CGA
X-BASIC調理実習/C調言語講座/X68000マシン語

- X1・MZ-2000/2500用RPG The Cave of Dalk
- うわさの68040，ついに登場

LIVE in '90　バーニングフォース(OPMD対応)

THE SOFTOUCH The Fille Professor/HOST PRO-68K

全機種共通システム　ファジィコンピュータシミュレータI-MY

▶前回の「言わせて……」では，MSX2+に浮気したといってましたが，今年はついにX1turbo2と離婚，そしてX68000EXPERTと再婚という暴挙にでてしまいました。X68000とは浮気のつもりだったのに。許してね X1turbo II。

主藤　二裕 (22) X68000EXPERT，MSX2+　福岡県

●編集室へのメッセージ

今月号の特集について

いちばん良かった記事

興味のなかった記事

これから載せてほしい記事内容

本誌以外にお読みのパソコン雑誌

推薦する市販ソフト

ソフト名：

推薦理由：

23ページの予告についてご意見をお聞かせください

あなたの愛機は(所有機種に○印をつけてください)　ない
X1(マニアタイプ,C,D,F,G,twin)　X1turbo(model 10,20,30,40,II,III,Z,ZII,ZIII)
MZ-(80K/C, 1200, 700, 1500, 80B, 2000, 2200, 2500, 2861)
X68000(ノーマル, ACE, PRO, EXPERT, [HD])　その他
FD(　基)　TAPE　QD　HD(　MB)　プリンタ(　)

年齢　歳　パソコン歴　年　男・女　プレゼントNo.

振替用紙

←点線から、きれいに切り取ってご使用ねがいます。

払込票

通常払込料金
加入者負担

口座番号	東京	1	-	29307
加入者名	株式会社 日本ソフトバンク			
金額	億 千 百 十 万 千 百 十 円			
払込人住所氏名	※			
備考		受付局日付印		

切り取らないで郵便局にお出しください。

記載事項を訂正した場合は、その箇所に訂正印を押してください。

払込通知票

通常払込料金
加入者負担

口座番号	東京	1	-	29307	金額	億 千 百 十 万 千 百 十 円
加入者名	株式会社 日本ソフトバンク				料金	払込み / 特殊 円
払込人住所氏名	※ (郵便番号)				備考	
					受付局日付印	

各票の※印欄は、払込人において記載してください。

この払込通知票は、機械で使用しますので、下部の欄を汚さないよう特に御注意ください　また、本票を折り曲げたりしないでください。　（郵政省）

この欄は、加入者あての通信にお使いください。
切り取らないで郵便局にお出しください。

切り取り線

送り先			性別	年齢	ご職業
	お名前	フリガナ	男・女		
	ご住所	フリガナ 〒			
	お電話	ご自宅	勤務先		

申込書				
THE COMPUTER 定期購読	新規申し込み 年 月号より	継続申し込み TC NO.	年間 7,200円
Oh.!PC 定期購読	新規申し込み 年 月号より	継続申し込み PC NO.	年間11,440円 6ヶ月 5,720円
Oh.!X 定期購読	新規申し込み 年 月号より	継続申し込み X NO.	年間 6,720円
Oh.!FM 定期購読	新規申し込み 年 月号より	継続申し込み FM NO.	年間 6,720円
月刊情報処理試験 定期購読	新規申し込み 年 月号より	継続申し込み J S NO.	年間 8,160円 6ヶ月 4,080円
C MAGAZINE 定期購読	新規申し込み 年 月号より	継続申し込み CM NO.	年間11,760円
パソコンマガジン 定期購読	新規申し込み 年 月号より	継続申し込み PM NO.	年間 6,960円
Beep メガドライブ 定期購読	新規申し込み 年 月号より		年間 5,760円
通信欄				

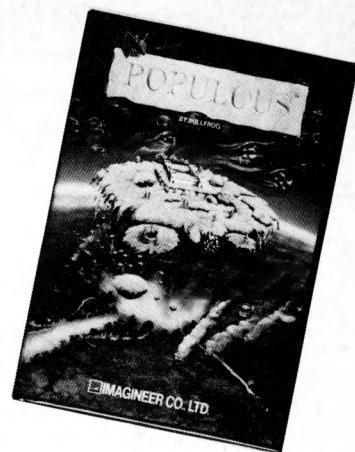
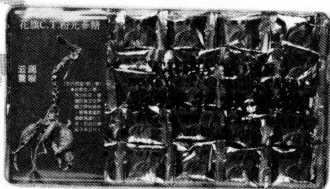
この払込通知票は、機械で使用しますので、下部の欄を汚さないよう特に御注意ください。また、本票を折り曲げたりしないでください。　(郵政省)

プレゼントの応募方法

3名

3名

3名

名

高麗人参飴

1名

毒物飲料の仕掛け人，古村(で)聡氏がわざわざアメリカのチャイナタウンまで行って買ってきた珍品。味はもちろん……。

1アルガーナ（神奈川県）藤原賢治（愛知県）米本剛（岡山県）山中泰徳　2レナム(東

川県)
井県)

――GAME OF THE YEARプレゼント当選者――

ゲームボーイ（静岡県）杉村謙一郎　**BATMOBILE**（山形県）築島啓介（東京都）堀端英彰（愛知県）藤井哲也（岐阜県）松久孝治（福岡県）立川智久　**RPG秘宝館**(東京都)中川比呂志（静岡県）大野二郎（三重県）服部靖司（広島県）土井一夫（大分県）首藤誠二　**RPG人名録**（宮城県）山田嘉明（埼玉県）丸山勝之（静岡県）高野真樹(愛知県)井戸浩登（兵庫県）土谷興正　**RPG100の疑問**（長野県）金子明人（大阪府）池水麦平中村宙史（福井県）宮本勝範（広島県）原田謙　**Oh! Xノート**（北海道）菅原克俊（千葉県）大谷伸介　吉岡哲（神奈川県）鈴木善昭（愛知県）稲葉貴也　福岡尚久(富山県)菅田朋樹（大阪府）山下智也（島根県）森星児（宮崎県）佐藤圭　他90名　（敬称略）

以上の方々が当選されました。おめでとうございます。商品は順次発送いたしますが，入荷状況などにより遅れる場合もあります。また，公正取引委員会の告示により，このプレゼントに当選された方は，この号の他の懸賞には当選できない場合がありますのでご了承ください。

（価格はすべて消費税別です）

▶ MIDIをやってみたいが，楽器もなければ，MIDIボードもない。やはり貧乏人は，FM音源までが限度なのだろうか。　渡部　裕亨（23）X68000EXPERT，MB-S1　福岡県

次世代マイクロプロセッサ登場

全盛を誇るRISCチップ

マイクロプロセッサの開発競争は相変わらず激烈で，新しい製品をフォローするのもたいへんです。そこで，最新のマイクロプロセッサ（主に32ビット）に関するデータを，日経エレクトロニクスの記事[1)2)]などから抜き出して表1にまとめてみました。この表には，チップ名，開発（製造）メーカー名，MIPS値,トランジスタ数，総命令数，キャッシュメモリ容量が示されています。ただ，MIPS値は設定方法などでかなり誤差がありますのでご容赦ください。

表は上下に分かれていて，上がCISCタイプ，下がRISCタイプのプロセッサです。乗算など複雑な命令を持たないRISCとCISCのMIPS値を単純に比較することは不公平なのですが，この表を見るかぎりRISC系は圧倒的にスピードが速いといえます。表には載ってませんがR6000（MIPS社）などは50MIPSを軽く越えているという話です。

CISCでは，相変らずインテルが8086系の石を力づくで作っているようで，80486などは命令数が214にものぼっています。RISCはコンパイラを作るのがたいへんだということですが，この大量の命令から最適な命令を選ぶのだってそれほど容易だとは思えません。また日本電気も，TRON陣営（表のGMICRO/200はTRONチップ）には加わらずに，Vシリーズで頑張っているのですが，RISC攻勢に耐えられるわけはなく，MIPS社のR2000/3000なども作っています。

今回焦点を当てるチップは単なるRISCではなく，次世代RISCの有力な方式として注目を集めつつある「スーパースケーラ方式」を初めて採用し市場にデビューしたインテルの80960CAです(80860のほうは別のアーキテクチャですので混同しないように)。

スーパースケーラ方式とは

計算機の高速化といってもさまざまなレベルの話がありますが，ここではプロセッサアーキテクチャについての話をします。プロセッサアーキテクチャの性能向上に関しては，「クロック周波数を上げることは素子技術的にかなり限界に近づいているので，あとは命令の実行を並列化するしかない」といわれています。やはり並列実行化がこれからの計算機アーキテクチャにとって必要不可欠というわけです。

汎用プロセッサの並列実行化として考えられたものとして，すでに紹介したVLIW方式や今回紹介するスーパースケーラ方式があるわけです。この方式の定義をアメリカの学会誌のコラム記事[3)]を参考にして短く書いてみると次のようになります。「スーパースケーラ方式とは，逐次的な命令ストリームから同時に実行できる命令をプロセッサが実行時に検出し，可能な場合には並列実行を行う方式のことである」。

ここでミソであるのが，「逐次的な命令ストリームから」というところです。いままでのプログラムは当然（並列実行ではなくて）逐次実行するように書かれているのですが，そのような逐次的な命令をプロセッサが並列実行してしまうということです。

したがって，理論的には従来のプログラムをそのまま並列実行できるというメリットがあるわけです。この点でトレーススケジューリングのような複雑な並列化コンパイラの問題があるVLIWのアプローチとは大きく違うところだといえます。原理的にコンパイル時にできるだけの並列化処理をしてしまうほうが，得られる並列度は大きいのですが，やはり互換性ということは予想以上に重要なことなのです。

インテルの80960CA

80960CAは，機械語命令をメモリから取ってくると，プロセッサ内の命令キャッシュにいったん格納します。それから4命令同時にデコードし，そこから同時に実行できる命令を最大3つ同時に実行開始します。ただし，3つの命令はどのような3命令でもいいわけでなくて，プロセッサ内の異なるユニットで処理を行う命令でないと駄目なのです。その種類は次の3つです。

1) REG形式

算術論理演算命令，レジスタ間コピー命令，比較命令など。

2) MEM形式

メモリとのロード，ストア命令，アドレッシングの計算だけを行うロードアドレス命令など。

3) CTL形式またはCOBR形式

分岐命令，手続きコール，リターン命令，ホールト命令など。

80960CAの命令アーキテクチャの特徴のひとつとして，静的な分岐予測を採用していることが挙げられます。高機能命令はプロセッサ内でマイクロコード列に展開されます。たとえば整数の比較を行うcmpi命令のあとにbne.t命令が書かれているとします。これはbranch if not equalということで，比較結果が等しくないのならどこかのアドレスに分岐するということです。bne.tのtはtrueを表しており，条件が成立して分岐する確率のほうが高いと，プログラムがプロセッサに教えているのです。

これにより，分岐が成立したときパイプラインが乱れ実行が遅れてしまうといったことが防げます。trueタグが条件分岐命令についていると，命令ストリームは分岐先のほうから順番に取ってきて命令パイプラインに入れるのです。なおtrueでないとき，つまりたぶん条件は成立しないだろうというときにつけるのはf(false)です。

また別の特徴として，このプロセッサは基本的にはRISCなのですが，アドレッシングの豊富さやマイクロコードで実現した一部の高機能命令などから，CISC的な色彩も感じとることができます。たとえば，高機能命令の変わったものとして，Conditional Compareというものがあります。これは文字どおり，条件が成立しているのならば比較をするというものです。つまりある変数xが，y1よりは大きくて，y2よりは小さいということをチェックするような処理の最適化のために用意されているようです[4)]。

表1　最新マイクロプロセッサ一覧

チップ	メーカー	MIPS	Tr数(万)	命令数	キャッシュ(KB)
80486	インテル	17	120	214	8
80386	インテル	8	27	141	*
68040	モトローラ	13.5	120	111	8
68030	モトローラ	12	30	76	0.5
32532	NS	10	37	129	1.5
V80	日本電気	13.1	93	119	2
V70	日本電気	4.3	38.5	119	*
GMICRO/200	日立,富士通,三菱	7	73	121	1
TX1	東芝	5-6	45	93	*
SPARC	Cypress	24	7.2	89	*
SPARC	B.I.T.	55	13	?	*
R3000	LSI Logic...	20	15	74	*
80860	インテル	35	100	76	*
80960CA	インテル	66	54.3	90	1
88100	モトローラ	28	16.5	51	*
29000	A.M.D.	17	20.8	115	*

▶最近，仕事が忙しすぎて，X1turboZを扱うヒマがないっ!! 夜遅く寮に帰ってきて，食事をして，風呂に入り，仕事のまとめをしたら，もう夜中。
磯部 政男 (21) X1turbo/turboZ，MZ-700 福岡県

80960CAの中身

まずイメージをつかんでもらうために，プログラムの実行の様子[5]を図1に示します。プログラムはメモリ上に通常の機械語プログラムと同じように順番に格納されています。図1の縦軸はその命令の並びで，横軸は時間軸を示しています。斜めの網掛けは演算のオペランドに対応するユニットへ出力するステップで，点の網掛けは各ユニットでの実行を示しています。

図1を見てわかるように，ロード命令(ld)と比較命令(cmpi)やロードアドレス(lda)命令と加算命令（addo）は同時に実行を開始しています。また，乗算命令の実行は4クロックかかるのですが，その終了を待たずに，加算命令などをオーバーラップして実行していることもわかります。

図2に80960CAの内部のブロックを示します。特にその核となる部分は，以下に示すような6つの部分に分けることができます。

1)　命令シークエンサ

4命令を同時にフェッチすることができます。フェッチした命令は内蔵する1Kバイトの命令キャッシュに格納し，4命令同時にデコードします。それから並列スケジューラで，4命令のうち同時に実行できる命令を調べ，最大3命令の実行を開始します。

2)　6ポートレジスタファイル

32ビットのレジスタを32個持っており，そのうち16個がグローバルレジスタで，16個がローカルレジスタです。6ポートであるということが重要です。1クロック内に6つの並列なアクセス（演算などのオペランド出力2つ，ひとつ前の命令の結果の格納，メモリへの出力，ひとつ前のメモリからの読み込み，5)のアドレス生成ユニットに対する出力）が独立にできます。

3)　整数実行ユニット

整数算術演算，論理演算，シフト，レジスタ間コピー，ビット演算，比較演算に関するすべての処理を行います。

4)　乗除算ユニット

乗除算，rem演算，mod演算を行います。乗算用に最適化されており，4クロックで終了します。3)の実行ユニットと並列実行します。

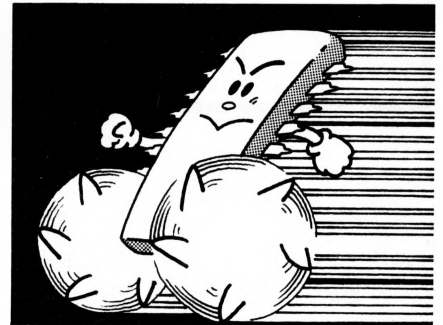
5)　アドレス生成ユニット

ロード命令やストア命令の実効アドレスの計算を行います。また，ロードアドレス命令を直接実行します。これは実効アドレスを指定したレジスタに格納する命令です。この命令があることで，ふつうのRISCプロセッサより複雑なアドレッシングを行うことができるのです。

6)　RAM/ローカルレジスタキャッシュ

1Kバイト分のスタティックRAMがメモリ空間の最初の部分にマッピングされています。残りの512バイトはプログラムからは見えませんが，ローカルレジスタのキャッシュとして使われています。コール命令，割り込み，ホールトにより，16個のローカルレジスタの内容は，このキャッシュの中に4クロックのスピードで自動的に退避されます[6]。もちろんあふれたら，外部のメモリ上のスタック領域に格納されます。

序の口なのだ

プロセッサアーキテクチャレベルにおける並列実行化の第一歩として，このスーパースケーラは大きな意味があります。従来のプロセッサでは，単に命令パイプライン

によって命令の投入間隔を短くして速度を上げるということが主流でしたが，スーパースケーラ方式では，複数の命令の実行開始（issue）を可能にしたという点で意味があるわけです。特に重要なのは，先述したようにコンパイラにしわ寄せがいかないということだと思います。

しかし逆にいえば，汎用プロセッサの並列実行化としては，ほんの序の口であるともいえるでしょう。そもそも原理的にはフォンノイマン型の逐次実行の概念自体を忠実に守っているからです。そして，そのようなアプローチでは，ふつうのソフトウェアの場合，2から3倍程度の速度向上しか望めないと多くの研究は語っているのです。

参考文献

1)〝特集：押し寄せるRISC〞，日経エレクトロニクス，No.474，pp.106-131，1989.

2)〝特集：どこへ行くCISCチップ〞，日経エレクトロニクス，No.476，pp.103-152，1989.

3)Michael Slater，〝VLIW，superscalar，and 64bit or not?〞，IEEE MICRO，1989-12，pp.103-104.

4)80960CAデータシート，Intel，1989.

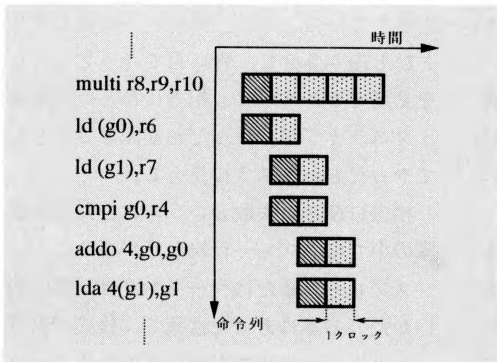
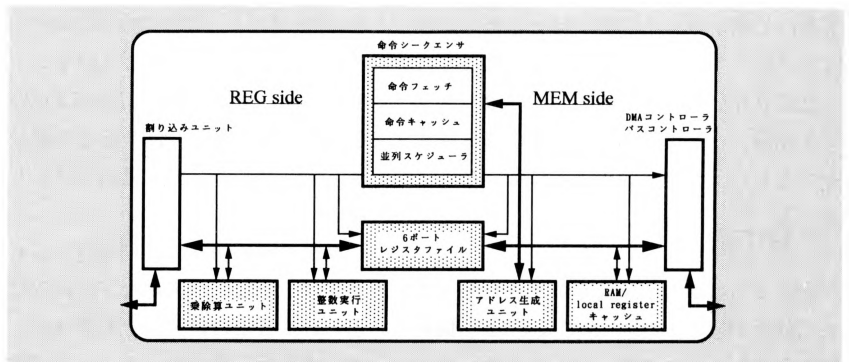
5)David Dannenberg，〝Superscalar技術を用い3命令の並列実行が可能な組み込み用マイクロプロセッサ〞，日経エレクトロニクス，No.490，pp.177-186，1990.

6)小栗清隆，〝インテルi80960CA技術解説〞，Computer Design，2月号，pp.86-93，電波新聞社，1990.

図1　並列実行の様子

図2　80960CAの内部構成

▶ X68000EXPERTとCZ-612Dをまとめて買いました。と，同時に1カ月前からOh！Xを買い始めました。これからヨロシク。　永江　幹伸（17）X68000EXPERT　佐賀県

第47回

猫とコンピュータ
開け！ ファイル

Takazawa Kyoko
高沢 恭子

突然のメッセージに大あわてのキョウコさん。さて，春になって，どうやらホンニャアも本来の元気さを取り戻したようです。今年もまたホンニャアの活躍が期待できそうですね。

「テンポラリーファイルがオープンできません!!」これなあに？　単語は1つひとつわかっても，伝えていることがわからない。いつものように，水道の蛇口をひねるような気軽さでディスクを立ち上げて，書きかけの原稿が入っているエディタのファイルを呼び出そうとした。「A>RED GENKOU 47」リターン。カッチンカッチンと，さあテキスト画面が出てきますよといったかいがいしい響きがするのだが，そのあと急に突きつけられたメッセージ。「テンポラリーファイルがオープンできません」

アラ？　こんなのいままで見たことがない。オープンできないって言われたって誰が悪いの？　ふだんと違うことが起こったらやりなおすのがいちばん。もういっぺんやってみたら解決したという経験は何回もある。そこでもう一度電源を入れるところからやってみるが，やっぱり同じだ。

原因を考えるよりやりなおすほうが簡単なので，なんべんもファイルネームを入れてみては，同じメッセージを見てムッとする私。あまりよく知らない相手と長いこと交際していて，いきなり思いがけないことを宣言されたみたいな気持ちだ。

これが「ディスクがいっぱいでなんにもできませんよ」と言ってくれたのなら，こんなに大あわてはしなかったのに。私も落ち着いて残りのバイト数に目をとめていればすぐに，ア，そうかとわかっただろうに。

ただそれだけのことで久しぶりのパニックを味わい，とうとう，出張先の夫に電話をするというナサケないことになった。

鉄の扉

前夜まではふつうに仕事ができたディスクに何かが起こった。「さては（夫が）何か妙な命令を入れたのだ。ひとこと言ってくれたらいいものを」。

「A> DIR」で見ると昨夜夫がNIFTYの通信ネットからダウンロードした異様に長いファイルがいくつかある。でも特に疑問のあるコマンドは見あたらない。

つぎにFD.EXEで A ディスクのすべての内容を調べてみるけれど，あやしいと思われる項目はない。

もしや，エディタのMIFESをREDにリネームして使っているのがいけないのかな。はじめに使っていたエディタがREDだったので，MIFESに変えてからも字数が少なくていいなんて，元の名前で使っている。MIFESに戻してみよう。でもダメ。そんなはずないなと思う。

このマウスが悪いのかな？　最近つけたマウスが何かしら妨害してるのかもしれない。そんな幼稚なことすら真剣に考えてしまう。

「テンポラリーがどうしたのよ」と心の中でつぶやくが，もうほんとにできることがなくなってしまったのだ。原稿どうしようかな。エディタがなくなっちゃうと手書きだ。いまさらそんな。

となりのNEC文豪7HRに目をやる。ワープロかぁ。電源を入れて少し動かしてみるけれど，初めてさわるようなもどかしさ。

X68000のワープロソフトは，もっと悠長なのだ。そういえばX68000のエディタもあったのだけど，使ったことがない。もう一度元に戻って，本棚からMS-DOSの入門書を何冊か抜き出し，何か応急処置を見つけようとしたが，とても見込みはありそうにない。

原稿の締め切りの時刻は迫ってくる。もうこれしかない。夫がたぶんいるだろうと思われるS市の研究所をめがけて，ついにダイヤルを回す。あっさりと夫が電話口に出て，ちょっとあきれたという声で，「それ，ディスクがいっぱいなんだよ，いくつかファイルをほかに移してごらん」。

ナサケない。ただそれだけのことか。見るとほんとに，残りのバイト数は1600ほどになっている。モオーッ!!　こんなことがわからないなんて。

それにしてもあのメッセージの神秘的なわかりにくさ。

こうしてついに,「猫とコンピュータ　第47回」の本文にめぐりあえた。

美しさのヒミツ

誰かが生まれつき足が速いとか，手先が器用だとか，記憶力がすぐれているといったようなことは，あまりこだわりなく人から認めてもらえる。でも，生まれつき美しいとか，魅力があるといったようなことになると，ほめる条件としてちょっと別のあつかいかたをされるみたいだ。

能力については堂々と競うのに，美しさの優劣を競うのは少し卑怯のように思われる。どちらも生まれついて持っている条件だし，それを輝かせるには努力がいる点で同じなのに。

ホンニャアは生まれたときから，目の中に2つの宝石を持っている。この澄んだ青い色と白い体のおかげで，彼はどんなにトクをしていることか。長いシッポをヒラリとひと振りさせて，青い目でキッとこちらを見据えるようすは，相当にひどい失敗やイタズラまでが，なんだか立派なことをしてやったというふうに見える。

相当にひどい失敗というのは，たとえば家の中でのスプレー行為などがある。

スプレー，またはマーキングは「猫の飼いかた」の本にあった言葉で，雄猫が自分のテリトリーを示すために要所にオシッコ

▶ X68000は決してゲームマシンではないっ！　僕はCコンパイラとアセンブラでそれを証明したい。
吉永　有輝（27）X68000，X1F　長崎県

をかけて回ることだそうだ。ホンニャアも一匹前になったころから，家の周囲のチェックポイットにスプレーを励行した。庭の四隅や物置，夫のゴルフバッグまでが標的になった。Ｓ市では，ブロック塀の上からわが家の植え込みに向けてスプレーするボス猫のアライグマと，境界線をかけた死闘を何回もくりひろげた。

このスプレーが有効なのは，強い臭いとその持続性だと思う。人間には耐えられないほどの悪臭だが，家の外でやっているぶんには雄猫の習性として認めるしかない。ところがうっかりした拍子にこれを室内でやってしまうのだ。

これは，はじめの段階ではホンニャアにとっては失敗ではなかった。家の外でやっていることを，室内でも必要と判断してやったのだから。でも，人間の生活の中では「猫のソソウ」なので彼はひどく叱られたのだ。このあたりの解釈が，猫にはとても難しい。建設的な作業のつもりだし，自分には芳香のスプレーだ。こんなによいことが，どうして叱られなければならないのか。庭ならばOKなのに，なぜ室内ではダメなのか。

もしそれがわかったとしても，動物の本能は反射的なものだ。柱とか，食器戸棚のそばを通るとき，自分でも気づかないような速さでスプレーしてしまう。やったとたんに，叱られた記憶がよみがえってシマッタと思い，あたりを駆けめぐったために誰かに悟られてしまうこともある。たとえ現行犯としてとらえられなくても，その有効な臭いの持続性のために，結果的には犯行場所に鼻先を押えつけられて，頭をペンペンと叩かれなくてはならないのだ。

こんな失敗も，ホンニャアの白い体と軽快な動作，青空のようなブルーの瞳の美しさのために，こちらも叱り声に迫力がないのがわかる。もし，おにぎりのような顔のアライグマや，全身がまん丸で，ひょうきんなデザインに仕上がっているミミが同じことをやったら，ずいぶんワリを食ってしまうだろう。ホンニャアは特に，青い目の魅力のためにトクをしている。

でも，ホンニャアの目がただ美しいだけでなく中から輝いているのは，鋭いカンの働きによる反応の確かさのためだ。ホンニャアはきっと，いつも自分をきたえるトレーニングをしているのだ。だから，２つの目が宝石になる。美しさと能力は，やはり分けられないものなのだ。

キャッツ・アイ

雨ばかりがつづいていた３月の上旬，すばらしいプレゼントのような，うららかな日曜日が訪れた。日本マイコンクラブ主催の「マイコン研究発表会」があるという日だった。

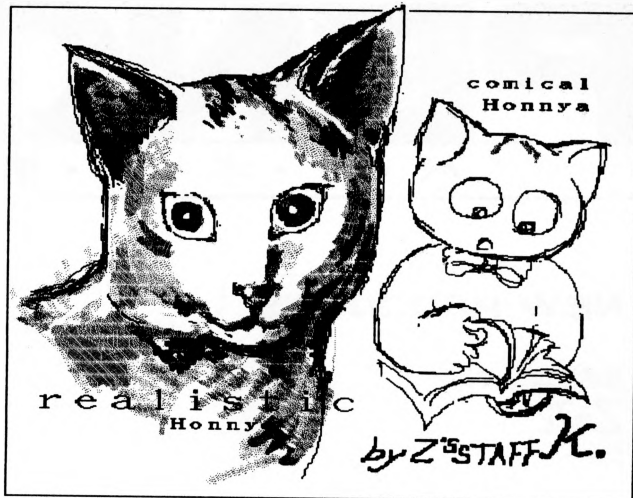
招待を受けたのは夫だが，作品が展示されるものと思って私もいっしょに出かけてみた。会場は東京タワーの向いにある機械振興会館の中の一室で，今年は第７回めだそうだ。

「研修室」という教室ふうの部屋で，16点の作品がつぎつぎに発表されていくが，展示やデモはないのでちょっと失望した。出品者は工科系の学生さんたち（個人やゼミのグループ）が多く，大学の先生や企業の中の著名な方もいる。

「電子楽器を用いた周波数特定システム」「言語発達遅滞児の為の語彙検査ソフトの開発」「ADコンバータを用いた交流回路の電圧，電流測定」「三太郎への文字修飾属性の変換」など，タイトルもアカデミックなら，聞いている60名ほどの方たちの熱心なこと。

私がいちばんよく理解できたのが，東京職業訓練短期大学校のグループの方たちの作品「テレビゲーム時の眼球運動」。

アイマークレコーダを使ってテレビゲーム時の眼球の動きを調べながら，ゲームの上手，下手と眼球運動の関係を調べる。このデータから，ゲームの難しさとは何か，使いにくいとはどんなことかを調べ，逆にソフト一般の使いやすさとはどうあるべきかを考える資料にもしたいそうだ。

20歳前後の男女10人に，任天堂のファミリーコンピュータで，シューティングゲーム「ザナック」を10分間やってもらう。その間アイマークレコーダを装着して，ビデオ収録する。10分間の中で継続時間が最長のものを解析データにして，AD変換し，PC-9801RAに取り込む。それにより眼球運動の特徴をあらわすパラメータの値を求める。

パラメータの値とビデオ観察から，上手な人と下手な人の違いを調べていく。複数の敵が画面にあらわれたときが，上手，下手の差がはっきりする。上手な人は前方の敵を攻撃しながら，側方の敵をかわす。目を移す前，攻撃か回避かの判断が的確で，目を移したらかならず攻撃する。下手な人はこの逆である。

敵を攻撃したあと，すぐにつぎの敵に移るのが上手な人で，いつまでも敵にとどまっているのが下手な人だそうだ。上手な人は移動がゆるやかで，敵に停留している時間が短く，下手な人はこの逆になる。

もうひとつの「ナンバーリングゲーム」による分析では，数値で見る難しさと，被験者の主観的な難しさの関係にも触れて，なかなか興味深かったが，説明上の省略が多くて残念だった。ゲームの難易度は心身の負担の大小も作用すると思われるので，今後はユーザーの生理，心理面との関連も測定に加えていくそうだ。

こうしたデータの結果そのものは，私たちの予想とそれほど異なるものではなかったけれど，マイコンを使ってメンタルな分野を解析してみるという試みは，なかなか実験的だと感じた。

「猫の目のように」というたとえは変化がめまぐるしいときに使われるが，猫がそれだけすばやく多様に反応する結果が，目にあらわれるということだろう。ホンニャアの目にアイマークのカメラを取り付けたら，どんな動きを示すのか。自分でアイマークを追いかけて，踊りだしてしまうかな。

▶「CYBERNOTE PRO-68K」の家計簿が動かない。メーカーはソフトのチェックをしているのだろうか？　などと思ってしまう。　坂本　祐治（34）X68000　熊本県

PENGUIN INFORMATION CORNER

ペ・ン・ギ・ン・情・報・コ・ー・ナ・ー

NEW PRODUCTS

低価格コピーマシン
Z-72
シャープ

Z-72

シャープは，B4サイズの原稿がコピーできる低価格パーソナル複写機「Z-72」(165,000円）を発売した。最大用紙サイズはA4で，B4原稿をコピーする場合はA4へ縮小する。縮小は２段階(0.8，0.7倍)，拡大は1.24倍のみ。給紙はカセット以外に手差しもできる。オプションで，赤，青，茶，緑のモノカラー現像カートリッジも発売される。低価格であることから小規模オフィスなどでの利用に適している。

〈問い合わせ先〉

シャープ㈱　☎06(621)1221，03(260)1161

カラー静止画TV電話
テレパシー LU-C10P
三菱電機

テレパシー LU-C10P

三菱電機は，業界初の電話回線用カラー静止画TV電話「テレパシー LU-C10P」(158,000円)を発売した。２種類のデータを位相を変えて同時に送信するQAM (Quadrature Amplitude Modulation) という新技術を採用し，モノクロタイプの２倍の高速通信が可能になった。標準モードの画像（100×160）で約８秒，高精細モード（200×320）は約25秒で送信できる。表示には４インチTFTアクティブマトリクス液晶ディスプレイを採用。カメラは本体から取り外して撮影することもできる着脱式（ケーブル１m）。また，TV，ビデオ，電子スチルカメラなどと接続することも可能。画像メモリは標準モード７枚分持っており，上書きを防止するロック機能もある。

〈問い合わせ先〉

三菱電機㈱　☎03(218)3134

立体物をコピー
ダ・ビンチ
キングジム

ダ・ビンチ

キングジムは，世界初の電子プリントカメラ「ダ・ビンチ」(49,800円）を発売した。デジタルカメラとモノクロプリンタを画像処理コンピュータのもとで一体化したもので，大きさは197×74×35mm (350g) の手の平サイズ。撮影した写真はすぐにプリントアウトできるがデータとして保存することはできない（記憶は１画面分のみ)。撮影した写真に対して，拡大，反転，輪郭線化などの画像処理を行うこともできる。カメラは露出2.8F，最短距離1m(オプションでマクロレンズ（3,800円）あり）で，セルフタイマー，逆光撮影なども可能。プリンタは解像度８ドット/mm，幅56mmのサーマルラインプリンタを搭載している。

〈問い合わせ先〉

㈱キングジム　☎03(864)1234

手書き入力方式電子ノート
IN-5000
キヤノン販売

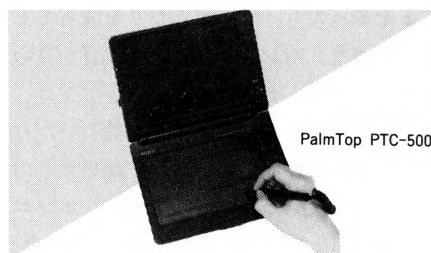
IN-5000

キヤノン販売は，AI支援の電子ノート「IN-5000」(75,000円）を発売した。付属のペンで画面にひらがな，カタカナ，数字などを書き込むだけで高度な情報管理を行える。手書き文字の認識にはキヤノン開発の人工知能技術「手書き認識Ai」を採用した。そのほかに機能としては，書き込んだ計算式の答えが得られる手書き計算機能，手書きワープロ機能，作画機能，世界時計機能，カレンダー・スケジュール機能，住所録管理機能なども持っている。また，RS-232Cインタフェイスを持っているためほかのパソコンやワープロとデータのやり取りもできる。同時に30種類の機能を持ったICカード「ノート拡張カード」(14,000円）も発売される。

〈問い合わせ先〉

キヤノン販売㈱　☎03(455)9681

個人情報を管理
PalmTop PTC-500
ソニー

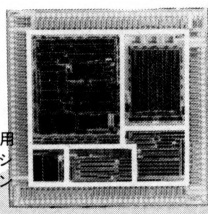
ソニーは，手書き入力方式の小型コンピュータ「PalmTop PTC-500」(198,000円)

PalmTop PTC-500

▶簡単に「X68000を買った」という人がいるけど，どこにそんな金があるのだろうか？
堀川　英雄 (20) X1turbo　大分県

を発売した。付属のペンで画面に直接書いた文字をデータとして入力することができる。認識できる文字は漢字を含め3535種類(内部データは7066種類持っている)。文字解析にファジィ理論を応用したため多少のクセ字でも読み取れる。さらに内蔵のマイク/スピーカーで8秒の音声メモも記録できる。搭載CPUはMC68HC000 (8MHz), ユーザーズメモリはデータ用, ワーク用それぞれ320Kバイト。ソフトウェアとして, 簡易ワープロ, スケジュール管理, 個人データ管理の3機能を持つ「PalmNote」が標準で付属する。同時にFAXアダプタ (70,000円), プリンタインタフェイス (20,000円), 2インチFDD (50,000円), メモリカード (16,000円) も発売。今後ICカードで専用ソフトが提供される予定である。

〈問い合わせ先〉
ソニー㈱ ☎03(448)3311

Z80高速化される
シャープ

Z80コマを用いたこのシステムオンチップ

シャープは, Z80系CPUコアの新製品2種類を発売する。電圧3V, 消費電力3mWの低電圧・低消費電力製品とクロック周波数25MHz, 最大アドレス空間18Mバイトの超高速製品の2種類 (従来製品は5V電圧, 消費電力45mW, 最大周波数8MHz, 最大アドレス空間は64Kバイト)。超高速型は, 命令処理の2クロック化 (従来は3, 4クロック)を内部データ幅の16ビット化(従来は8ビット) と1クロックで演算結果を出力できる高速ALUを採用することにより実現した。価格は, 低電圧型は従来と同等, 高速型は従来の3倍程度になる予定である。

〈問い合わせ先〉
シャープ㈱ ☎06(621)1221, 03(260)1161

X68000とMS-DOSをリンク XIN/XOUT
データスペックジャパン

データスペックジャパンは, RS-232Cを介してデータ転送をするシステム「XIN/

XIN/XOUT

XOUT」(7,800円) を発売した。バイナリファイルの転送も可能で (エラーチェック方式は独自のものを採用), ワイルドカードも使用できる。パッケージはRS-232Cケーブルとファイル転送プログラム(3.5インチ2DD, または5インチ2HD) から構成されており, MS-DOSおよびPC-DOSマシン全機種とX68000で使用できる (DOSは入っていない)。また, BBSでのオンラインユーザーサポートも行う。

〈問い合わせ先〉
データスペックジャパン㈱ ☎03(774)7741

INFORMATION

夢のクリスタルギャラリー
シャープ

クリスタルギャラリー

シャープ東京ショウルームで5月31日まで「夢のクリスタルギャラリー」が開催されている。「液晶のシャープ」が誇る液晶関連製品を揃えたもので, 試作品や参考製品などが目白押しだ。

展示は, 昼はショウウィンドウとして夜はスクリーンとして使える液晶ミラクルスクリーン。110インチの液晶ハイビジョンシステム。高精細液晶ビジョンシステム。50インチ液晶ディスプレイを9枚組み合わせた液晶150インチマルチビジョン。家庭用の液晶リビングシアター。ビジネス向けの液晶AV会議システム。オーディオマニア向けの液晶AVマニアルーム。山小屋風の液晶ロッジシアターなど。

ほとんどのものは客が実際に体感でき液晶製品の醍醐味が味わえる。展示ブースはコース別に分かれており, それぞれでシャープの女性社員が説明してくれる。展示品の大部分は, 高価な販売用の製品やここでしか見ることができない参考製品であるので, 東京近郊の方はぜひ一見を。

〈問い合わせ先〉
シャープ㈱ ☎03(260)1161

CGAコンテストビデオ配布
DōGA

DōGAは,「第2回アマチュアCGAコンテスト」の入選作品を収めたビデオテープの有料配布を行う。入選12作品, 作品解説, オープニングアニメーションを収めたVHS60分テープ。希望の方は下記の要領で申し込むこと。なお, 手違いが起こらないように下記の項目は厳守してほしい。条件を満たしてない場合, 申し込みに応じられないこともありえる。

1) 申し込み期間:1990年4月18日～5月31日 (当日消印有効)
2) 申し込み方法:郵便振替のみ (ただしカンパは除く)
3) 口座番号:大阪3-109598 (加入者名DOGA)
4) 配布価格:(2,000円)+カンパ (任意)
5) 払い込み用紙には, 住所, 氏名, 連絡先の電話番号を必ず明記すること。また, 深夜まで帰宅しなかったり出張などで留守にすることが多い方は, 確実に受け取れる住所 (勤務先, 友人宅など) を書く。
6) 通信欄 (通常裏面にある) には,「CGAコンテストビデオ希望」と明記すること。また, 金銭以外のカンパがある場合通信欄の空いたスペースに「カンパ○○を別送」と書いてほしい。
7) カンパは金銭でなくともよい。米などの食料, 各地の名産品などは大歓迎 (ただし, ビール券は最近ダブついているとのことである)。金銭以外のカンパの場合, プロジェクトルームに直接送ること。その場合でも実費2,000円分は郵便振替とする。

なお, 申し込みが終わった時点からダビングを開始するため, 実際に発送が開始されるのは7月以降になる予定である。

〈問い合わせ先〉
プロジェクトチームDōGA
〒533 大阪市東淀川区淡路5-17-24 102号

▶アフターバーナー, う～ん私には単調すぎて退屈してしまう。いまいちばんのお気に入りはファーストクィーンです。あんまり宣伝してないから売れてないみたいだけど。でも, 当然, 発売と同時に買って3カ月はまりっぱなし, いいゲームなら宣伝しなくてもきっと売れる!
尾薗 明彦(23) X68000ACE-HD宮崎県

FILES Oh!X

このインデックスは，タイトル，注記——筆者名，誌名，月号，ページで構成されています。春眠，暁を覚えずとはよくいったもので，寝過ごしてしまうこともしばしば。皆さんはどんな春，迎えてますか？

参考文献
I/O　工学社
ASCII　アスキー
コンプティーク　角川書店
テクノポリス　徳間書店
POPCOM　小学館
マイコン　電波新聞社
マイコン BASIC Magazine　電波新聞社
LOGIN　アスキー

一般

▶ネットワーカー・ホリック第17回
オンラインで遊べるマルチゲームネット「ゲームネット恒星紀」や，絵の出る通信版対局将棋・囲碁を扱ったネット「J&P HOTLINE」を紹介。——編集部，LOGIN，5号，204-205pp.

▶ The News File！
データグローブで音楽を演奏する！　1月28日から4日間，渋谷の Bunkamura で開かれたハイパーインストゥルメントコンサートの模様を紹介。その他 TRON やエプソンの話題など。——編集部，LOGIN，6号，26-31pp.

▶コンピュータウィルスの恐怖
コンピュータウイルスとは何か？　症状は？　侵入形態は？　など，一般人にはわかりにくいその姿を解説。また感染の予防についても説明している。——編集部，マイコン BASIC Magazine，4月号，59-66pp.

▶なんでも Q&A
AX286D のクロック切り換え法，キーボードやフロッピーインタフェイスの接続，MS-DOS での日付の表示法などの質問に答える。——編集部，マイコン4月号，376-377pp.

▶ミッション・ザ・ムーン
米国のアポロ計画を LD の映像と HyperCard からなるマルチメディアデータベースがパイオニア LCD 社から発売。その内容と試みについて報告する。——謝緋子，ASCII，4月号，422-423pp.

▶これからの CAI を考える
CAI 登場の背景やその可能性，将来について，教育統合ソフト，CAI への応用が注目される DVI 技術などについて詳細にレポートする。——編集部，マイコン，4月号，131-154pp.

▶警視庁交通管制センター
車のあふれる東京の道路を信号操作や情報提供で管制する警視庁のシステムを取材。——野沢潤一郎，マイコン，4月号，166-169pp.

▶気ままに PSG 講座
最終回。PSG でのパーカッションと効果音を出す方法について。——はちみつ川野俊充，マイコン，4月号，228-232pp.

▶たかが水力発電，されど水力発電
長野県の七倉ダムに水力発電の現在をみる。電力需要に応じた発電を行うのにコンピュータ管理を導入。——菊池秀一，マイコン，4月号，279-283pp.

▶電子温度計を作る
アナログ/デジタルのインタフェイスによく使われるコンパレータを使って電子温度計を製作してみよう。——石川至知，マイコン，4月号，354-357pp.

▶じゃんけんマシーンの製作
音声再生ボードを利用してじゃんけんの相手を女性の声でしてくれるモノを作る。——米田敏文，I/O，4月号，261-263pp.

MZ-80K/C/1200/700/1500

MZ-1500（MZ-5Z001 BASIC）

▶嗚呼！　大仏
大仏を動かして善人を助け，悪人をやっつける。白熱のアクションゲーム。——まてりある，マイコン BASIC Magazine，4月号，129-130pp.

MZ-80B/2000/2500/2800

MZ-2500（M25-BASIC）

▶戦ひませう
気合いを溜めて相手にぶつかり勝負する。オーソドックスな2人用アクションゲーム。——白井俊明，マイコン BASIC Magazine，4月号，131-132pp.

X1/turbo/Z

X1シリーズ

▶誌上公開質問状
X1G の BASIC のコピーの方法を解説。——多田太郎，マイコン BASIC Magazine，4月号，89p.

▶オニゴッコ X1
あなたは矢です。タイムオーバーまでにモンスターをやっつけましょう。——堀田英克，マイコンBASIC Magazine，4月号，154-155pp.

▶ GOD OF GAMER
3つのゲームで得点を稼ぎ，その総合得点でランクを上げて GOD OF GAMER を目指す。——ズオ，マイコン BASIC Magazine，4月号，156-157pp.

▶ The Map Editor X1
ロールプレイングや迷路ゲームに使うマップを作成するためのユーティリティ。——Mr.Octopus，マイコン BASIC Magazine，4月号，174-175pp.

X1turbo シリーズ

▶月刊ソーサリアンニュース
ギルガメッシュ・ソーサリアンの攻略ポイント，ボスキャラを紹介。次回発売予定のセレクテッド・ソーサリアン3も少しだけ紹介。——編集部，テクノポリス，4月号，30-31pp.

▶アメリカゲーム特集
4月発売予定のウィザードリィVを紹介。——編集部，コンプティーク，4月号，90-92pp.

▶ HOW TO WIN

カミソリのように鋭利で研ぎ澄まされた文体。ひとつところに留まらない鮮やかな流れ。単一の時系列にとらわれない展開。アンダーグラウンドの作家ウィリアム・バロウズの小説である。あのディックを原作とする映画とはなんの関係もない，ブレードランナーという架空の映画のシナリオを語る小説だったりする。バロウズのなかでは読みやすい小説だが，1つひとつ意味を追おうとすると楽しくない。細かい意味はすっとばして，ひたすら鋭利な文体を味わいながら流れにのっかるのが正しい読み方。バロウズを読んでいると（といっても，私はほかの作品はあっさり挫折したうえアメリカのシーンにはまったく疎いのだが），サイバーパンク文体の原点を見た気がする。ストーリーのほうはといえば，2014年のニューヨーク，アンダーグラウンドに潜った医者と薬の運び屋（こいつがブレードランナー）の物語である。「現代医学の奇跡は，自然の免疫機構に介入することで，長期的に見ると予防するより多くの病を生みだすことになる」といった真実に感じてノルのがいい。（K）

映画：ブレードランナー　ウィリアム・S・バロウズ　山形浩生訳　リブロポート
☎03(983)6191　B6判　77ページ　1,236円

書籍の価格は消費税込みです

ギルガメッシュ・ソーサリアンの攻略法を掲載。——編集部，コンプティーク，4月号，112-115pp.

▶ SHOVEL KUN

シャベルくんの使命は土砂を運んで穴を埋めること。穴を全部埋め終われば面クリア。コンストラクション付き。——駒井健也，マイコンBASIC Magazine，4月号，158-159pp.

▶最新ゲーム徹底解剖!!

ソーサリアンシナリオ集であるギルガメッシュ・ソーサリアンの舞台，グラフィックを紹介。——編集部，LOGIN，6号，170-173pp.

X68000

▶ SOFT RADAR

発売予定のパズルゲームのプロディアや，綱引きゲームのタッグ・オブ・ウォーを紹介。——編集部，POPCOM，4月号，8-19pp.

▶ WE ARE THE X68000 WORLD

新着ゲームのワンダラーズフロムイース，ジェミニウイング，グラナダ，スーパーハングオン，キューブランナー，バブルボブルを紹介。——編集部，POPCOM，4月号，80-83pp.

▶先取りおすすめゲーム

新着ゲームの紹介。ヨーロッパで人気のシミュレーションゲーム，ポピュラスやダンジョンマスターを紹介する。——編集部，テクノポリス，4月号，7-15pp.

▶ GAMING WORLD

新着ゲームのサンダーブレードや，発売予定のクォース，グランディフロラム，グラナダ，ジェミニウイング，SLIMYER，タッグ・オブ・ウォー，キューブランナー，サークを紹介する。——編集部，テクノポリス，4月号，22-32pp.

▶ X68000新聞

4月発売予定のX68000用RPGサークや，ZERO～第4のユニット4～，ミュージ君やミュージ郎のデータが生かせるMusicstudio Mu-1といった新着ソフトの紹介。そのほか，X68000ユーザーのたまり場SPS-NETなど。PDSはファイルユーティリティのTFを紹介。——編集部，LOGIN，6号，118-121pp.

▶ NEW SOFT

3月発売予定の超人気シミュレーションゲームのポピュラスや，4月発売予定のシューティングゲームのジェミニウイング，そのほかアクションパズルのキューブランナーや，ワンダラーズフロムイースを紹介。——編集部，LOGIN，5号，12-19pp.

▶ X68000新聞

3月発売予定のワンダラーズフロムイースや，X68000オンリーのシューティングゲーム「グラナダ」を紹介。そのほかジェミニウイング，キューブランナー，サンダーブレード，HOST PRO-68Kなど。PDSは謎の常駐型環境ソフト，もろこしを紹介。——編集部，LOGIN，5号，132-137pp.

▶アメリカゲーム特集

洋モノの移植ソフトの中で，最近もっとも期待されているX68000版ポピュラスを紹介。そのほかダンジョンマスターなども紹介している。——編集部，コンプティーク，4月号，84-100pp.

▶ SPIRITS

ワンダラーズフロムイース，サンダーブレード，プロディア，銀河英雄伝説，グランディフロラム，グラナダを紹介。——編集部，コンプティーク，4月号，216-219pp.

▶誌上公開質問状

X68000の特殊画面制御とは？ MUSIC PRO-68K[MIDI]で内蔵FM音源を操作することはできるか？ などの質問に答えている。——多田太郎，マイコンBASIC Magazine，4月号，90p.

▶ Stone of Theory（理論の石）

落下してくるブロックを動かすのではなく，最下段のブロックをジョイスティックで左右に移動させてブロックを消す，テトリスもどきゲーム。——YAMAMO SOFT，マイコンBASIC Magazine，4月号，160-161pp.

▶ The Adventure of Hover Craft

ジョイスティックを2本も使用して，アサルトタイプのゲームを作成。ホバークラフトを操作してスペシャルフラグを10本取る。——高橋 潤，マイコンBASIC Magazine，4月号，162-164pp.

▶ビデオゲーム版テトリス

ゲームミュージックプログラム。——進藤慶到，マイコンBASIC Magazine，4月号，186-188pp.

▶ワンダラーズフロムイース for X68000

ついに移植された人気パソコンゲーム，ワンダラーズフロムイースを大紹介。他機種との違いやイースシリーズの今日までの歩みなど。——佐久間亮介，マイコンBASIC Magazine，4月号，225-228pp.

▶ X68000のPCMを使う

PCM音源の概要とHuman68k上での利用方法について。また，OS-9/X68000のAD PCMをはじめとした各ドライバの構造と動作についても解説する。——中山進，ASCII，4月号，333-340pp.

▶ TFS.X

X68000用常駐型ファイルセレクタ。ディレクトリやファイルをウィンドウに表示し，複数のファイルを簡単に選択できる。——大野洋史，ASCII，4月号，383-384pp.（リスト471-480pp.）

▶ AV STRASSE

Mu-1，サンダーブレード，JTeXの紹介・評価を行う。——編集部，ASCII，4月号，397-404pp.

▶ X68000マシン語入門

先月に引き続いて「OPMDRVを使って自分独自の音色を出す方法」を扱う。OPMの各パラメータについての解説。——高橋雄一，マイコン，4月号，196-202pp.

▶スーパーハングオン

シャープブランドで発売されたスーパーハングオンの評価記事。——あゆさわかつみ，マイコン，4月号，222-223pp.

▶なんでもQ＆A

CARD PRO-68Kでのくしざし計算法，その他ワープロ，VS，RAMディスクの活用における疑問に答える。——編集部，マイコン，4月号，374-375pp.

▶ SCS

PC-9801と兼用のソースリストのチェックサム表示プログラム。注釈行は計算しない工夫もされている。——WIZARD N氏，I/O，4月号，190-195pp.

▶ G98

PC-9801の640×400×8色のグラフィックをテキスト画面に表示するユーティリティ。——Zanobia，I/O，4月号，210p.

▶インタラプトチェッカ

割り込み状況を表示し，必要に応じて解除する。常駐型ソフトのON/OFFに便利。——西方茂樹，I/O，4月号，226-230pp.

▶ Mu-1

X68000のシーケンスソフトMu-1をレビュー。——L&M，I/O，4月号，292-293pp.

ポケコン

PC-1600K

▶ポケコン電子手帳

最終回。10回にわたって製作したポケコン電子手帳の総括を行う。——塚田洋一，マイコン，4月号，326-333pp.

PC-1360

▶ PC-100m走

2つのキーを交互に押してタイムを競う。1/100秒まで計ることができるんだぞ。——Yabusho，I/O，4月号，211p.

PC-E500

▶ ENONE

エノンを操作してゴールを目指す。ブロックに狭まれたら爆弾で道を開こう。パニックアクションゲーム。——佐藤祐紀，マイコンBASIC Magazine，4月号，169p.

▶とりでの攻防

おなじみのとりでの攻防，移植版。——小松賢治，マイコンBASIC Magazine，4月号，170p.

エデンの西（上）

ジョブズとスカリー。コンピュータにたずさわったことのある人なら，この2人の名を一度は耳にしたことがあるだろう。この本は，若くしてその地位を築いた彼ら2人とアップルの成長記録であり，そして彼らの情報革命十字軍理論とウォール街新時代論理との衝突の記録でもある。ちなみにこのタイトルは「アップルを追われたジョブズの落ち着く先は，カリフォルニア・ドリームをはらんだ“エデンの西”だった」ことからきている。

フランク・ローズ著　渡辺敏訳　サイマル出版会
☎03(582)4221　B6判　312ページ　1,800円

BBSの歩き方

最近にわかに注目を浴びているパソコン通信。取り立ててパソコンを扱い慣れていなくても，簡単に楽しめるのが人気の秘密だろう。ただ，そんな手軽なパソ通も，それぞれのBBSによってメニューやコマンドが異なっているため，なかなか馴染みづらいという欠点があるようだ。そんなときこの本を見れば，BBSのメニュー構成とコマンドがすぐにわかるようになっている。パソ通の初心者にもベテランにも便利な1冊だ。

SE編集部編　翔泳社　☎03(263)0447　A5判
204ページ　1,200円

書籍の価格は消費税込みです

編集室から from E・D・I・T・O・R

DRIVE ON

このコーナーでは，本誌年間モニタの方々のご意見を紹介しています。今回は，3月号の記事に関するレポートです。

●よく言われていることではあるが，いまのパソコンミュージックの状況は初期のパソコンCGのそれとよく似ている。本当にいいものを作るにはツール自体を作らないといけないからだ。その点から特集「ミュージックアドベンチャー」はよかった。現在の時点はこういった環境整備が最重要ポイントだと思うからだ。ただ，はっきり言っていまのMIDI環境は高価すぎて手が出せないという現状はある（最低レベルを揃えるだけで10万円ほどかかる)。そういった意味からも「なんでも鳴らせるOPMD.X」はMIDIを持っていない人にも使えるためいいと思う。

西田宗千佳（18）X68000，X1Fmodel 120 千葉県

●X68000にも音楽的な環境が整ってきたと思います。これは周辺機器やソフトの普及だけではなく一般ユーザーの間での意識の浸透なども意味します。それゆえ今回大幅な特集を組んだことは，その意識をさらに高める意味でグッドタイミングだったと思います。以前はMIDIといえばほかのマシンが中心であったのですが，現在店頭でもっとも耳を引く音を出しているのはX68000です。またコンピュータをAV機器として見ている人もいる時代です。そういったことから特集「ミュージックアドベンチャー」は一般の方にいままでの楽器とは異なった新しい電子音楽の可能性を与えるいいチャンスであったと思います。

特に「なんでも鳴らせるOPMD.X」は，MIDI対応ということで多くの人が利用するでしょう。一段と誌面を投稿者たちが賑わせてくれそうですね。MIDIユーザーは今後激増するはずですしこのようなサポートがあるのはとてもいいことだと思います。またOPMAフルコンパチということは優れたことであるし，ボスコニアンからデータを流用できることなどとても便利だといえるでしょう。なによりもそのデータの汎用性の高さは素晴らしいものです。また3月号は「Live in '90」や「X-BASICプログラミング調理実習」などからも音楽色の濃い号だという感じを受けますが，このように全体をひとつの雰囲気でまとめるのは非常にいいと思います。

大津和之（20）X1turboZ 福岡県

●特集のデバイスドライバOPMD.Xはとても強力なうえいままで使っていたものがそのまま使えるという点で評価できると思います。特にMIDIDRV.SYSをどこからでも使えるのが便利です。ところで「MIDIデータローダ＆セーバ」でX1用のMIDIボードの記事が出ているのには正直言って驚きました。2年前の記事なのに専用のソフトが掲載されるなど信じられません。自作派を中心としているというOh!Xの編集のポリシーをひしひしと感じ，嬉しくなりました。

湯澤聡（27）X68000，X1turboIII，MZ-2531/2861，MSX，PC-1360K，PC-6601 埼玉県

●MIDI環境はまだまだこれからだという気がしていますが。BEEP音のみだったのがPSG，FM音源と変わってきて，MIDIが急速に普及するのも間近のような気もします。特集「ミュージックアドベンチャー」で紹介されたプログラムは皆実用的で使えるものだと思いました。特にOPMD.XはOPMA完全コンパチでMIDI＋FM＋AD PCMの同時演奏が可能というのでかなり実用に耐えるものになっていると思います。

田中実（19）X68000ACE，X1turboII 大阪府

●今回の「DōGA・CGアニメーション講座」は映像論という趣向の変わったものであった。しかし，本来こういった解説は連載中で並行して行うべきだったと思う。アニメにとっての命はやはりテーマを伝えるためのバランス感覚ではないかと思い，そのための技術は必要だからだ。今後もこういった映像表現のテクニックについての解説はやったほうがいいだろう。

中野賢一（30）X1/G/turboII，MZ-2000，FM-8，PC-9801ES5/N，FP-200，PC-1251，B16LX 山口県

●今回のS-OSで発表された「超多機能アセンブラOHM-Z80」は，マクロ命令，分割アセンブル，ファイルのインクルード，条件アセンブラなどを備えたなかなか優れたマクロアセンブラであると思います。実際に使用していないので使用感はよくわからないのですが，速度的にはどうなのでしょうか？　機能的には，マクロや構造化制御文が強力なのは十分評価できると思います。また，リファレンスマニュアルを見た限りでは文法も通常のザイログアセンブラと同じで理解しやすくコマンドモードやアセンブルモードの強力さが目を引きました。今後もこのコーナーが活性化することを期待します。

森川一（24）X68000ACE-HD，X1turboII 北海道

ごめんなさいのコーナー

4月号 The Cave of Dalk

P.121 MZ-2000の場合，グリーンディスプレイへの対応に不備がありました。

グリーンディスプレイを使用する方は，まずリスト1のあたまに，

9CF8 3E 0C CD F4 1F C3 64 96

の8バイトを新たに加え，リスト6の以下のアドレスをそれぞれ，

9601 64 → F8
9602 96 → 9C
96EC DB → C9
96F7 DB → C9
9729 D0 → 55
972D D0 → 55
99B5 D0 → 55

のように変更してください。

また，MZ-2000/2200/2500，X1/turboの各機種ともテープを利用する場合に不都合がありました。テープユーザーの方は，まずリスト1のあたまに，

9CF0 CD 15 C4 CD 02 A1 C9

の7バイトを新たに加え，以下のアドレスをそれぞれ，

CCC6 15 → F0
CCC7 C4 → 9C
CCCC 15 → F0
CCCD C4 → 9C
CD25 15 → F0
CD26 C4 → 9C

のように，変更してください。

バグに関するお問い合わせは
☎03(230)7683(直通)
月〜金曜日16：00〜18：00

お問い合わせは原則として，本誌のバグ情報のみに限らせていただきます。入力法，操作法などはマニュアルをよくお読みください。

また，よくアドベンチャーゲームの解答を求めるお電話をいただきますが，本誌ではいっさいお答えできません。ご了承ください。

▶X68000のCMをぜひ中畑清さんにやってもらいたい。
加藤　真哉（19）X68000EXPERT-HD，MZ-2200　鹿児島県

次号は創刊8周年 特別付録は期待大！ 雨天順延？

▼Oh！Xは次号で創刊８周年を迎えますが，それを記念し，X68000のあっと驚く実用プログラムディスク「創刊８周年記念PRO-68K」が付録に付くことになりました。本誌ではかねてから誌面には載せられないような大きなプログラムやユーザーの皆さんにぜひとも持っていてもらいたいツールやシステムをどうやって供給すればよいか考えてきたのですが，ようやくその第１弾が実現することになったわけです。雑誌にディスクが付くのはもう珍しくもありませんが，Oh！Xがやる以上，掲載プログラムを収録しただけのものや，広告を兼ねたデモディスクではなく実用本位の強力なものをお届けする予定です。ただそのため次号は特別定価ということで，なんと220円も高い780円となる予定です。今回はX68000ユーザー以外の方には申し訳ないのですがどうかご容赦ください。

▼今月号の特集「BASICプログラミング」はいかがでしたか。今回はプログラム作成への取り組み方にポイントをおき，初心者レベルからちょっとハイレベルまで段階的に記事を用意しました。Oh！Xではひとりでも多くの方にもっとプログラミングの楽しさを知ってもらいたいと思っています。ご意見をお寄せください。

▼Oh！Xでは新しくスタッフを大募集いたします。仕事の内容は，原稿の執筆，プログラム開発，投稿作品のチェックなど多岐にわたりますが，時間の拘束などはありません。応募資格は東京近郊にお住まいの社会人および学生(高卒以上)でOh！Xの誌面作りに参加したい人。希望者は，住所・氏名・年齢・電話番号を明記のうえ,自己PR(投稿経験があればそれも）などを含めた自由論文を6000字以内（本誌約２ページ分）にまとめ，Oh！X編集部「スタッフ希望」係までお送りください。お待ちしています。

▼「C調言語講座PRO-68K」はページの都合により今回はお休みになりました。申し訳ありません。

投稿応募要領

- 原稿には，住所・氏名・年齢・職業・連絡先電話番号・機種・使用言語・必要な周辺機器・マイコン歴を明記してください。
- プログラムを投稿される方は，詳しい内容の説明，利用法，できればフローチャート，変数表，メモリマップ（マシン語の場合）に，参考文献を明記し，プログラムをセーブしたテープ（ディスケット）を添えてお送りください。また，掲載にあたっては，編集上の都合により加筆修正させていただくことがありますのでご了承ください。
- ハードの製作などを投稿される方は，詳しい内容の説明のほかに回路図，部品表，できれば実体配線図も添えてください。編集室で検討の上，製作したハードが必要な場合はご連絡いたします。
- 投稿者のモラルとして，他誌との二重投稿，他機種用プログラムを単に移植したものは固くお断りいたします。

あて先

〒102 東京都千代田区九段南2-3-26井関ビル
日本ソフトバンク出版部
Oh!X「テーマ名」係

SHIFT・BREAK

▶某所では，攻撃的な奴と大物ぶっている奴とのうすっぺらい他機種批判論争が続いていた。と思ったら表層的平和論者たちが出てきて鎮めてしまった。コンピュータの用途は個人の勝手だとは知っているが，自分たちがそれに我慢できないマニアとは気づかずに，青筋立てながら理性的なフリをしている。面白い奴らだ。　（やっちゃえやっちゃえ派H.U.）

▶３月も中盤をすぎたので，武道館ではどこぞの大学の卒業式シーズンまっさかりであった。おかげで九段下駅ではダフ屋のおっちゃんがいなくて静かだなあ。ところで，ついに私も年貢のおさめどきが来たようだ(といっても結婚するわけじゃない)。つまりX 68000を買うだけの話だ。でも私の場合はX1パワーアップのためのX68000なのだ！　（亀）

▶先日アメリカ旅行に行ってきました。TVを見ていて思ったのですが，海軍の隊員募集のCMが異常にカッコいい！　さすがトップガンの国，などと変な感動をしてしまいました。イメージいいでしょうねー，あれなら。さて，Oh！Xでもスタッフ募集中なのですが，うちのスタッフってどんなイメージなんでしょ？　怖くないから来てくださいね。（で）

▶最近ビリヤード場の数が減ってきているが，ビリヤードが趣味の私にとってはとても悲しい。でも値段が安くなってきたから，悪いことばかりじゃないんだけどね。流行って本当にあっという間に過ぎ去ってしまうんですね。最近いちばん嬉しかったことといえばスーパーハングオンでEXPERTを制覇したことぐらい。あー，これって悲しいなあ。（H.K.）

▶カーCDというのはあるのに，二輪用は見かけない。ヘルメットの中にヘッドホンを仕込むのは，道交法に照らすとマズいのだそうだが，最近買ったポータブルCDをこっそり載せて走ってみた。さすがに振動がひどいのでときどき音が飛ぶ。電源もバイクから取ったのにいささか惜しい。高速道路では安定するかな？　こんど試そっと。（A.T.）

▶言葉によるコミュニケーションは難しい。たとえば「だから」という接続詞。自分の「だから」と相手の「だから」が一致していないと，たちまちにして「説明不足の慌て者」か「順序立てて説明できない間抜け」か「自分が世界標準だと思っている思い上がり野郎」になる。だから，言葉によるコミュニケーションは難しい(さて)。（Mu）

▶さもなくば，すべての人間はモルモットである。ラッシュには順応した。合成食品には順応した。花粉症になった(ささやかな抵抗)。体内に蓄積された異物に過密と過労働でどこまで耐えられるか。ないものを流通させてないものを得るシステムにどこまで従順か。思ったよりみんな素直。これも教育の成果。私は単なる落ちこぼれ(ラッキー)。（K）

▶先月の編集後記を書いた直後にⅣはあっけなく終わった。だが，同時並行してプレイしたⅢを終わらせるにはさらに１週間を要した。やはり最高傑作はⅢだと思いながら，パーティを変えて最後の敵を２度倒したあとは，レベル99の遊び人４人ではどうかとレベルアップに励む今日この頃。うーん，遊び人はレベルが上がるほど役に立たなくなる。（KO）

▶いま靖国神社は雨で散った桜をサカナに夜な夜な宴会まっ盛り。気分だけでも，と仕事サボってぼ～っと桜を見てたら，タコ焼き屋の兄ちゃんがいきなりタコ焼きをくれた。内心またか（私はこーいう場でほんっとによくモノをもらう）と思った瞬間，私は悟ったね。私はテキ屋好みの顔なんだと！　さあ，目指すは鬼龍院花子のワールドか!?（E.O.）

▶たしかに映画は金にならない。なかでも自主映画には貧乏の香りが深く染み込んでいる（某製作所社長談)。フィルムはビデオに駆逐され続けた。そしてついにエクタクロームは製造中止になった。市場の論理とはいえこれは寂しい。寂しい理由はビデオに比べて８mmにも優れたところがあるからではない。メディアが制限されていくことだ。（S）

▶ユーザーインタフェイスの未来。おじいちゃんおばあちゃんでも誰でも間違いなく簡単に扱えるコンピュータがある未来と，複雑なOSをものともせずキーボードに指を走らせるおじいちゃん，おばあちゃんのいる未来ではどっちがいいんだろうか？　さて，女性からのプレゼントは無条件にうれしいです。しかし，百年の孤独……とは……。（U）

▶椅子を並べてガーガー寝ているとなにやら話声が。もうろうとした頭で起き上がると見慣れない輪郭の人物が座っている(私の視力はかなりファジーだ)。しかし誰かを確認しないまま再び意識が遠のいてしまった。そうか，君だったのか。遙々来てくれたのに申し訳ない。例のお土産はなかなかよくできているよ。（冷麺同好会というのはすごいと思うT）

▶セガMEGA-DRIVEのときは「ようこそ！」だったが，SNKのNEO-GEOはどうなんだろう？　おそらく「こりゃなんだ!?」でしょうね。　清水　義弘（21）X68000　沖縄県

microOdyssey

コンピュータで制御された部屋に観客がいる(データスーツのようなものを着ているのだろうか?)。部屋で観客に映像を見せる。観客はその映像を見て反応を示す。するとコンピュータが観客の反応に再度反応し別の映像を作成し映し出す。つまり観客とコンピュータがお互いにフィードバックを繰り返すというものだ。とある本でこのような実験が紹介されていた。

この実験はリプチンスキーなどビデオ作家の作品を彷彿させるものがある。当然のことであるが映像が目指しているものも文章などと同じく「真実」を表現することだ。そして(非常に乱暴な表現ではあるが),ビデオにおける真実追求へのアプローチは「特撮」と「どっきりカメラ」の2つに集約される。前者は映像を構成する写真自体を本物らしくリアルにし,後者は設定によって映像に真実味を与えようとするものだ。ところで映像における「真実表現」のための技術は,映像メディアの「デジタル化」と「インタラクティブ化」が実現されたら飛躍的に向上するのではないだろうか。

もし完全にデジタル化が実現されれば,何度ダビングしても画像が劣化しない→何重もの合成が可能→ブッシュがゴルバチョフにパンチを食らわせる「実写の写真」が作れる。これはヘルやサイテックスなどのイメージWS(3000dpi,A1サイズ程度のフルカラー画像)を使えばいまでも可能だ。これらの画素数はもはや印刷物やフィルムの解像度を越えている。この技術を駆使すれば(リプチンスキーの)「イマジン」程度はデスクトップで作れるはずだ。

またインタラクティブな映像メディアができたら,ブッシュがゴルバチョフに電話する→ゴルバチョフは朝起きたばかりで顔が腫れている→リアルタイムに合成された映像に肉声を重ねながら送る,といったことが可能だ(声を合わせるタイミングが難しいけど)。これは大量のデータを送る回線があれば解決する。光ファイバーでは毎秒5億ビットのデータ転送ができるので,毎秒30コマ程度は楽勝だ。

「この100年間,人々は写真に写っているものは現実に存在するものだと思い続けてきた,しかしそういう時代は歴史的に見て例外的なのだ」という意見がある。同感だ。本来メディアとは加工でき,加工した部分は作った人以外には気づかれてはいけないのだ。写真の編集(修整)は情報操作を目的とした「偽物作り」のためではなく「本物作り」のために行われなければいけないし,「嘘はいくらでもつけるけど嘘をついてはいけない」という文章では当たり前のことが映像メディアにも適応されなければいけないのだ。これを実現してくれるのはデジタル技術だ。放送局ではクロマキーを使った映像合成が当然のことのように行われているが「デジタルの嘘」にはかなわない。

このような映像環境がパーソナルなレベルで実現したらどうなるだろう? きっと手書きの文章が本になるように,自宅で作曲した音楽がレコードになるように,個人が作った映像作品をそのまま市場に流せるようになるのではないだろうか(「編集」できるようになるからだ)。この映像における「Art Decade」の到来はそんなに遠くないような気がする。 (S)

1990年6月号5月18日(金)発売

Oh! Xは8周年だ! 特別定価780円の謎と賢い定期購読の方法とは? アンケート用紙の奥に秘められた**Oh! X読者の生態**にも迫るぞ。**S-OSは5周年だ!** ついにベールを脱ぐ**PC-286対応S-OS"SWORD"**(互換機可)と本当に動くのか**X68000対応S-OS"SWORD"**! その他,今度こそ始まる**ハード工作入門**,謎の**X1 turdo用コマンドシェルシミュレータ**(仮題)など。とにかくめでたいぞ!

バックナンバー常備店

東京	神保町	三省堂神田本店5F	03(233)3312
	〃	書泉ブックマートBI	03(294)0011
	〃	書泉グランデ5F	03(295)0011
	秋葉原	T-ZONE 7Fブックゾーン	03(257)2660
	八重洲	八重洲ブックセンター3F	03(281)1811
	新宿	紀伊国屋書店本店	03(354)0131
	高田馬場	未来堂書店	03(200)9185
	渋谷	大盛堂書店	03(463)0511
	池袋	リブロ池袋店	03(981)0111
	〃	西武百貨店9F コンピュータ・フォーラム	03(981)0111
神奈川	横浜	有隣堂横浜駅西口店	045(311)6265
	〃	有隣堂ルミネ店	045(453)0811
	藤沢	有隣堂藤沢店	0466(26)1411
神奈川	厚木	有隣堂厚木店	0462(23)4111
	平塚	文教堂四の宮店	0463(54)2880
千葉	柏	新星堂カルチェ5	0471(64)8551
	船橋	リブロ船橋店	0474(25)0111
	〃	芳林堂書店津田沼店	0474(78)3737
	千葉	多田屋千葉セントラルプラザ店	0472(24)1333
埼玉	川越	黒田書店	0492(25)3138
	川口	岩渕書店	0482(52)2190
茨城	水戸	川又書店駅前店	0292(31)0102
大阪	北区	旭屋書店本店	06(313)1191
	都島区	駸々堂京橋店	06(353)2413
京都	中京区	オーム社書店	075(221)0280
愛知	名古屋	三省堂名古屋店	052(562)0077
	〃	パソコンΣ上前津店	052(251)8334
	刈谷	三洋堂書店刈谷店	0566(24)1134
長野	飯田	平安堂飯田店	0265(24)4545
北海道	室蘭	室蘭工業大学生協	0143(44)6060

定期購読のお知らせ

Oh!Xの定期購読をご希望の方は,とじ込みの振替用紙の「申込書」欄に何年何月号からをご記入のうえ,**年間購読料6,720円**(税込)を添えてお申し込みください。その際,裏面の通信欄に「○年○月号よりOh!X定期購読希望」と忘れずに明記してください。なお,すでに定期購読をご利用いただいている方には,購読期限終了と同時にご通知申し上げますので,同封の払込用紙をご利用ください。

海外送付ご希望の方へ

本誌の海外発送代理店,日本IPS(株)にお申し込みください。なお,購読料金は郵送方法,地域によって異なりますので,下記宛必ずお問い合わせください。

日本IPS株式会社
〒101 東京都千代田区飯田橋3-11-6
☎03(238)0700

 5月号

■1990年5月1日発行 定価560円(本体544円)
■発行人 孫 正義
■編集人 橋本五郎
■発売元 (株)日本ソフトバンク
■出版事業部 〒102 東京都千代田区九段南2-3-26 井関ビル
Oh!X編集部 ☎03(230)7681
出版営業部 ☎03(230)7670 FAX 03(262)8397
広告センター ☎03(297)0181
■印 刷 凸版印刷株式会社

日本ソフトバンクの 書籍特約書店

下記の書店の一覧は、日本ソフトバンク書籍特約店として右にある商品の他、新刊もとりそろえております。ご希望の商品がある場合は、下記のお近くの書店にてお買い求め下さい。

(注)現品が売れて補充中の場合もございますので、ご注意下さい。

SOFT BANK

日本ソフトバンク出版事業部

〒102 東京都千代田区九段南2-3-26 ☎03(230)7670

全国特約書店一覧

地域	書店名	電話
〈北海道〉		
札幌市	紀伊國屋書店札幌店	011-231-2131
〃	旭屋書店札幌店	011-241-3007
〃	丸善札幌支店	011-241-7252
〃	リーブルなにわ	011-221-3800
〃	冨貴堂札幌パルコ店	011-214-2303
〃	ダイヤ書房本店	011-712-2541
〃	ダイヤ書房西店	011-655-6223
旭川市	旭川冨貴堂	0166-26-3481
〃	ブックス平和マルカツ店	0166-23-6211
苫小牧市	旭屋書店苫小牧店	0144-36-5185
〈東　北〉		
青森市	成田本店	0177-23-2431
〃	岡田書店	0177-23-1381
弘前市	紀伊國屋書店弘前店	0172-36-4511
〃	ブックイン城東	0172-28-2882
八戸市	伊吉書院	0178-44-1917
盛岡市	東山堂書店本店	0196-53-6464
〃	さわや書店	0196-53-4411
〃	第一書店	0196-53-3355
仙台市	金港堂	022-225-6521
〃	金港堂ブックセンター	022-223-0979
〃	アイエ書店駅前店	022-264-0718
〃	丸善仙台支店	022-266-1127
〃	高山書店	022-263-1511
〃	ブックスみやぎ	022-267-4422
秋田市	三浦書店	0188-33-8131
山形市	八文字屋	0236-22-2150
福島市	岩瀬書店コルニエツタヤ店	0245-21-2101
〃	博向堂	0245-21-1161
郡山市	東北書店	0249-32-0379
いわき市	ヤマニ書房本店	0246-23-3481
〃	鹿島ブックセンター	0246-28-2222
会津若松市	宝文館	0242-27-5198
原町市	文芸堂	0244-22-1720
〈関　東〉		
水戸市	川又書店駅前店	0292-31-0102
〃	ツルヤブックセンター	0292-25-2711
勝田市	武石書店	0292-73-1212
東海村	大野書店	0292-82-2098
鹿島郡	なみき書店	0299-96-1855
土浦市	共栄堂	0298-21-6134
つくば市	丸善筑波大学会館店	0298-51-6000
〃	友朋堂吾妻本店	0298-52-3665
宇都宮市	落合書店オリオン店	0286-34-3777
〃	落合書店東武ブックセンター	0286-34-8271
〃	新星堂宇都宮店	0286-33-2337
小山市	進駸堂駅ビル店	0285-25-1522
前橋市	煥乎堂	0272-23-1211
〃	リブロ前橋店	0272-34-1011
〃	戸田書店前橋店	0272-61-5063
高崎市	学陽書房	0273-23-4055
〃	サカ井書店	0273-62-1500
〃	新星堂高崎店	0273-27-3961
〃	戸田書店高崎店	0273-63-5110
太田市	ナカムラヤ	0276-22-2001
〈首都圏〉		
浦和市	須原屋本店	048-822-5321
浦和市	須原屋コルソ店	048-824-5321
大宮市	押田謙文堂	048-641-3141
〃	ブックセンター押田	048-647-3141
〃	三省堂ブックポート	048-646-2600
蕨市	須原屋蕨店	0484-44-1211
川口市	岩渕書店川口店	0482-52-2190
川越市	黒田書店川越店	0492-25-3138
所沢市	芳林堂所沢店	0429-25-5355
〃	いけだ書店所沢店	0429-28-3271
上福岡市	黒田書店上福岡店	0492-66-0120
朝霞市	文教堂朝霞店	0484-76-0107
志木市	新星堂志木店	0484-74-0182
春日部市	文教堂春日部店	048-752-7666
比企郡	錦電サービス	0492-96-2962
千葉市	多田屋セントラルプラザ店	0472-24-1333
〃	キディランド千葉店	0472-25-2011
習志野市	巌翠堂	0474-72-5011
船橋市	ときわ書房本店	0474-24-0750
〃	リブロ船橋店	0474-25-0111
〃	旭屋書店船橋店	0474-24-7331
〃	芳林堂津田沼店	0474-78-3737
〃	第二巌翠堂	0474-65-0926
〃	三省堂書店西船橋店	0474-34-3111
柏市	西口アサノ	0471-44-2111
〃	新星堂柏店	0471-64-8551
松戸市	堀江良文堂	0473-65-5121
〃	辰正堂駅ビル店	0473-64-7997
横浜市	有隣堂トーヨー店	045-311-6265
〃	有隣堂東口ルミネ店	045-453-0811
〃	栄松堂相鉄ジョイナス店	045-321-6831
〃	そごうブックセンター	045-465-2111
〃	丸善ブックメイツポルタ店	045-453-6811
〃	有隣堂伊勢佐木店	045-261-1231
〃	有隣堂戸塚店	045-881-2661
〃	文華堂戸塚店	045-864-5151
〃	アーバン文華堂	045-821-5151
〃	文教堂青葉台南口店	045-983-5150
川崎市	有隣堂アゼリア店	044-245-1231
〃	有隣堂川崎BE店	044-200-6831
〃	文学堂本店	044-244-1251
〃	文教堂溝ノ口店	044-811-8258
鎌倉市	島森書店大船店	0467-46-3841
〃	鎌倉書店	0467-46-2619
横須賀市	平坂書房WALK店	0468-25-5537
藤沢市	有隣堂藤沢店	0466-26-1411
〃	リブロ藤沢店	0466-27-0111
〃	文教堂六会店	0466-82-9610
茅ヶ崎市	川上書店ルミネ店	0467-87-3827
平塚市	サクラ書店駅ビル店	0463-23-2751
〃	文教堂四之宮店	0463-54-2880
小田原市	八小堂書店	0465-22-7111
〃	伊勢治書店	0465-22-1366
〃	文教堂小田原店	0465-36-3677
厚木市	有隣堂厚木店	0462-23-4111
大和市	文教堂中央林間店	0462-75-4165
相模原市	文教堂相模大野店	0427-49-0650
〃	文教堂橋本店	0427-74-5581
相模原市	文教堂星ヶ丘店	0427-58-6121
津久井郡	文教堂城山店	0427-82-9278
〈東　京〉		
千代田区	三省堂書店神田本店	03-233-3312
〃	書泉グランデ	03-295-0011
〃	東京堂書店	03-291-5181
〃	旭屋書店水道橋店	03-294-3781
〃	丸善お茶の水店	03-295-5581
〃	巌翠堂	03-291-1362
〃	いずみ神田南口店	03-254-8521
〃	明正堂秋葉原店	03-257-0758
〃	T-ZONE	03-257-2660
中央区	八重洲ブックセンター	03-281-1811
〃	日本橋丸善	03-272-7211
〃	旭屋書店銀座店	03-573-4936
港区	書原新橋店	03-591-8738
〃	雄峰堂ＮＳ店	03-503-6586
〃	虎ノ門書房本店	03-502-3461
〃	虎ノ門書房田町店	03-454-2571
品川区	芳林堂大井町店	03-474-4946
〃	明屋書店五反田店	03-492-3881
渋谷区	紀伊國屋書店渋谷店	03-463-3241
〃	旭屋書店渋谷店	03-476-3971
〃	三省堂書店渋谷店	03-407-4545
〃	大盛堂書店	03-463-0511
〃	紀伊國屋書店笹塚店	03-485-0131
新宿区	紀伊國屋書店本店	03-354-0131
〃	三省堂書店新宿西口店	03-343-4871
〃	福家書店センタービル店	03-345-1246
〃	福家書店野村ビル店	03-342-0298
〃	新星堂ＮＳビル店	03-344-2055
〃	西武新宿ブックセンター	03-208-0380
〃	芳林堂高田馬場店	03-208-0241
〃	未来堂	03-200-9185
豊島区	旭屋書店池袋店	03-986-0311
〃	芳林堂池袋店	03-984-1101
〃	リブロ池袋店	03-981-0111
〃	三省堂書店池袋店	03-987-0511
〃	新栄堂本店	03-984-2345
〃	新栄堂アルパ店	03-988-0181
台東区	明正堂中通り店	03-831-0191
墨田区	ブックストア・談	03-635-1841
葛飾区	文教堂青戸店	03-838-5938
江戸川区	文教堂西葛西店	03-689-3621
大田区	アクトブックスサンカマタ店	03-735-1551
〃	竜文堂大森駅ビル店	03-775-3851
中野区	明屋書店東京本社	03-387-8451
杉並区	ブックセンター荻窪	03-393-5571
〃	書原杉並店	03-313-4778
武蔵野市	紀伊國屋書店吉祥寺東急店	0422-21-5543
〃	弘栄堂吉祥寺店	0422-22-1031
〃	パルコブックセンター吉祥寺	0422-21-8122
調布市	真光書店	0424-87-2222
府中市	啓文堂	0423-66-3151
三鷹市	三省堂書店三鷹店	0422-48-4510
〃	東西書房	0422-46-0275
小金井市	文教堂小金井店	0423-86-0161
国分寺市	三成堂国分寺店	0423-25-3211

展示図書一覧

定価は本体価格です。

- MS-DOSいたれりつくせり本 ●1800円
- プレイMS-DOS ●1900円
- UNIX System V プログラマ・ガイド ●12000円
- UNIX System V ユーザ・ガイド ●9800円
- UNIXオペレーティングガイド ●3000円
- OS/2 APIブックⅠ ●2709円
- C言語の活用理解 ●2000円
- C言語の基礎知識 ●2500円
- C言語の応用50例 ●2300円
- 上級・C言語の応用例50例 ●2400円
- Cプリプロセッサ・パワー ●2200円
- Play the C 上・下 ●各1500円
- Turbo C入門 ●2600円
- C++プログラミング ●2600円
- Quick Cプログラミング ●2602円
- 詳説C言語 ●4369円
- 8086アセンブリ言語 ●2800円
- 8086マクロプログラミング ●2600円
- Final Ver.4.0ブック ●2400円
- MIFES Ver.4.0ブック ●2400円
- ビジネスソフトデータ活用ブック ●2800円
- BASICによるプログラミングスタイルブック ●1800円
- ソーティング・ノート ●1900円
- J-3100パワーユーザーブック ●2400円
- 続・PC工作入門 ●1800円
- PC-286Lブック ●1700円
- 試験に出るX1 ●2800円
- RDBファラオ活用ガイド ●2903円
- 言図ガイド ●2301円
- Rydeenガイド ●2427円
- P1EXEガイド ●2524円
- Lotus1-2-3ガイドⅡ ●2500円
- MS-Chart Ver.3.1ガイド ●2900円
- まいと～くガイド ●2300円
- 新松ガイド ●2000円
- 一太郎Ver.3ガイド ●2500円
- 新一太郎ガイド ●2300円
- 桐Ver.2ガイド ●2500円
- 花子応用ガイド ●2500円
- Lotus 1-2-3ガイド ●2400円
- P1ガイド ●2300円
- Ninja2ガイド ●2300円
- Multiplan Ver.3.1ガイド ●2400円
- アセンブラCASL入門 ●2000円
- ハードウェア徹底マスター ●2500円
- FORTRAN徹底マスター ●2800円
- 情報処理の基礎知識 ●1600円
- COBOL徹底マスター ●2900円
- 受験用語ハンドブック ●1800円
- 情報処理入門1・2 ●各1204円
- CASLで学ぶアセンブラ言語入門 ●2204円
- バイト&ワードの風にのって ●1800円
- 田原総一朗のパソコンウォーズ ●1400円
- パソコンを襲う知的独占の戦い ●1600円
- RPG幻想事典・日本編 ●1800円
- 魔法王国シムルグント ●1800円

国立市	東西書店	0425-75-5061
小平市	文教堂小平店	0423-43-9229
東村山市	文教堂東村山店	0423-96-1115
立川市	オリオン書房ウイル店	0425-27-2311
八王子市	くまざわ書店本店	0426-25-1201
町田市	有隣堂町田店	0427-23-3018
〃	久美堂本店	0427-25-1330
〃	久美堂小田急店	0427-27-1111
〃	文教堂鶴川店	0427-35-4117
〃	文教堂小川店	0427-96-1781
多摩市	くまざわ書店桜ヶ丘店	0423-37-2531
福生市	文教堂福生店	0425-53-7708
〈甲信越・北陸〉		
甲府市	文教堂甲府店	0552-22-4600
長野市	平安堂長野店	0262-26-4545
〃	長谷川書店	0262-26-2122
上田市	平安堂上田店	0268-22-4545
松本市	ブックスロクサン	0263-35-5555
〃	改造社松本駅ビル店	0263-36-3777
飯田市	平安堂飯田店	0265-24-4545
岡谷市	笠原書店	0266-23-5070
諏訪郡	平安堂下諏訪店	0266-28-1111
新潟市	紀伊國屋書店新潟店	025-241-5281
〃	萬松堂	025-229-2221
〃	北光社	025-228-2321
長岡市	覚張書店	0258-32-1139
〃	ブックセンター長岡	0258-36-1360
〃	長岡技大長峰文化	0258-46-6437
上越市	パソトピア コスモス	0255-25-5867
山北町	BOOKメディア	0254-77-3850
富山市	瀬川書店	0764-24-4566
〃	清明堂	0764-24-4166
〃	BOOKSなかだ豊田店	0764-32-1353
〃	文苑堂本郷店	0764-22-0552
〃	文苑堂赤江店	0764-33-0321
高岡市	文苑堂	0766-21-0333
〃	文苑堂横田店	0766-21-0431
金沢市	うつのみや片町店	0762-21-6136
〃	書林香林坊本店	0762-20-5011
野々市町	王様の本本店	0762-46-5325
福井市	勝木書店	0776-24-0428
〃	品川書店新田塚店	0776-24-1112
〈東海〉		
静岡市	静岡谷島屋呉服町本店	0542-54-1301
〃	江崎書店	0542-54-4481
〃	吉見書店	0542-52-0157
〃	戸田書店SBS店	0542-81-5733
〃	戸田書店曲金店	0542-81-5899
沼津市	吉野屋	0559-23-5676
〃	マルサン書店宝塚店	0559-63-0350
富士市	戸田書店富士店	0545-51-5121
清水市	戸田書店本店	0543-65-2345
浜松市	浜松谷島屋連尺本店	0534-53-9121
名古屋市	三省堂書店名古屋店	052-562-0077
〃	星野書店近鉄ビル店	052-581-4796
〃	丸善名古屋支店	052-261-2251
〃	丸善ブックメイツセントラルパーク	052-971-1231
〃	日進堂上前津店	052-263-0550
名古屋市	三洋堂パソコンショップΣ	052-251-8334
〃	三洋堂いりなか本店	052-832-8202
〃	ちくさ正文館本店	052-741-1137
〃	白樺書房西店	052-774-7223
豊橋市	精文館	0532-54-2345
岡崎市	ブックス鎌倉	0564-54-1822
豊田市	三洋堂梅坪店	0565-35-2334
豊川市	三洋堂豊川店	05338-3-0334
刈谷市	三洋堂刈谷店	0566-24-1134
春日井市	三洋堂勝川店	0568-32-7806
岐阜市	自由書房	0582-65-4301
大垣市	大洞堂ブックス258	0584-81-2553
〃	大洞堂岐大バイパス店	0584-74-7766
一宮市	三洋堂一宮店	0586-77-5734
可児市	三洋堂可児店	0574-63-2334
多治見市	三洋堂多治見店	0572-24-0340
津市	別所書店11ビル店	0592-24-1014
四日市市	文化センター白揚	0593-51-0711
鈴鹿市	シェトワ白揚スズカ	0593-82-5221
〈近畿〉		
京都市	駸々堂京宝店	075-223-1003
〃	アバンティ・ブックセンター	075-682-5031
〃	オーム社書店河原町店	075-221-0280
〃	ジュンク堂京都店	075-252-0101
〃	オーム社書店竹田店	075-644-2611
奈良市	駸々堂大丸店	0742-26-6241
大阪市	旭屋書店本店	06-313-1191
〃	紀伊國屋書店梅田店	06-372-5821
〃	オーム社書店大阪店	06-345-0641
〃	駸々堂京橋店	06-353-3209
〃	駸々堂心斎橋店	06-251-0881
〃	旭屋書店ナンバ店	06-644-2551
〃	ナンバブックセンター	06-644-5501
〃	ヒバリヤ書店ナンバ店	06-644-5407
〃	旭屋書店アベノ店	06-631-6051
〃	ユーゴー書店	06-623-2341
〃	河村書店	06-951-2968
枚方市	水嶋書房京阪デパート店	0720-51-3432
高槻市	コーベブックス西武高槻店	0726-83-1766
東大阪市	ヒバリヤ書店本社	06-722-1121
神戸市	ジュンク堂センター街店	078-392-1001
〃	ジュンク堂サンパル店	078-252-0777
〃	海文堂書店	078-331-6501
〃	日東館書林	078-391-8701
姫路市	新興書房	0792-85-3344
〃	誠心堂書店	0792-81-2055
和歌山市	宮井平安堂	0734-31-1331
〃	帯伊書店	0734-22-0441
〈中国〉		
岡山市	紀伊國屋書店岡山店	0862-32-3411
〃	丸善岡山支店	0862-31-2261
津山市	津山ブックセンター	08682-6-4047
広島市	紀伊國屋書店広島店	082-225-3232
〃	丸善広島支店	082-247-2251
〃	金正堂	082-248-3715
〃	積善館	082-248-3151
尾道市	啓文社尾道店	0848-37-5151
福山市	啓文社福山店	0849-22-3111
福山市	ブックシティ啓文社	0849-25-0050
〃	啓文社コア	0849-41-0909
山口市	五十部誠文堂	0839-24-6630
〃	文栄堂	0839-22-5611
下関市	中野書店	0832-22-6181
宇部市	京屋書店	0836-31-2323
〃	末広書店	0836-31-0086
防府市	誠文堂国衙店	0835-25-1988
光市	三文字屋	0833-71-0251
鳥取市	富士書店	0857-23-7271
松江市	園山書店	0852-21-4167
〈四国〉		
徳島市	小山助学館本店	0886-54-2135
〃	小山助学館東口店	0886-25-1380
〃	森住丸善	0886-23-3228
高松市	宮脇書店本店	0878-51-3733
丸亀市	宮脇書店丸亀店	0877-22-5533
松山市	紀伊國屋書店松山店	0899-32-0005
〃	明屋書店本店	0899-41-4141
〃	明屋書店大街道店	0899-41-4242
〃	丸三書店	0899-31-8501
新居浜市	明屋星原店	0897-44-4000
宇和島市	明屋宇和島店	0895-23-1118
高知市	金高堂	0888-22-0161
〈九州・沖縄〉		
福岡市	紀伊國屋書店福岡店	092-721-7755
〃	りーぶる天神	092-713-1001
〃	積文館新天町店	092-781-2991
〃	福岡金文堂本店	092-741-2106
〃	福岡金文堂朝日ビル店	092-431-1094
〃	福岡金文堂デイトス店	092-451-6175
〃	福岡金文堂アニマート原	092-844-0088
北九州市	ナガリ書店	093-521-1044
〃	金栄堂	093-531-3685
〃	旭屋書店北九州店	093-631-6421
〃	井筒屋ブックセンター	093-641-0131
〃	カルパーク平野	093-661-7988
〃	白石書店本城店	093-601-2200
久留米市	エマックスたがみ	0942-33-1841
飯塚市	BOOKリード	0948-25-7266
大分市	パルコブックセンター大分店	0975-35-0643
〃	本町晃星堂	0975-33-0231
別府市	明林堂	0977-23-2183
宮崎市	中央、田中書店	0985-24-3511
〃	寿屋宮崎店	0985-27-4111
佐賀市	金華堂北バイパス店	0952-32-1965
〃	積文館佐賀店	0952-24-4314
〃	積文館デイトス店	0952-23-7155
長崎市	メトロ書店	0958-21-5453
〃	好文堂	0958-23-7171
佐世保市	金明堂書店	0956-22-4214
熊本市	紀伊國屋書店熊本店	096-322-5531
〃	長崎書店	096-353-0555
人吉市	明屋人吉店	0966-22-5486
鹿児島市	春苑堂ブックプラザ	0992-25-3200
〃	ブックスみすみ	0992-57-1011
那覇市	球陽堂書房ビル店	0988-63-3752
〃	文教図書	0988-62-1201

今、X68000の通信が変わる!!!

た〜みのる2

ユーザー重視の機能を搭載して

好評発売中

17,800円

X68000 専用

パソコン通信ソフト

「た〜みのる2」はX68000用に製作された通信ソフトです。X68000の機能を充分に引き出して、ユーザーの方々が簡単に操作できるよう工夫・製作されています。

24/31KHz ディスプレイ対応

「た〜みのる」が装いも新たに「た〜みのる2」として登場！
「た〜みのる」が通信入門版なら
「た〜みのる2」はマニアタイプの通信ソフトです!!

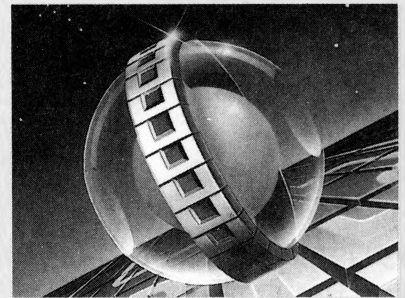
〈機能概要〉

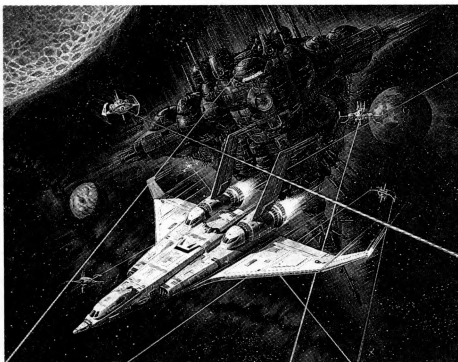
★ウインドウメニュー方式による機能選択。★オートダイヤル・オートログインプログラムの自動作成機能。★オートログインプログラムのユーザー作成可能。★「た〜みのる2」起動時オートダイヤルするホストの設定が可能。(登録により起動時指定ホストへのオートダイヤル可能)★アップロード・ダウンロード機能。★アップロード時のウエイト種別の選択、及び各ウエイト時間の設定機能。(文字間待ち時間・行間待ち時間・待ち文字列の設定)★XMODEM方式(SUM128/CRC128/CRC1024)によるアップロード・ダウンロード機能。★バックログ(受信バッファ)機能。(直接送信・保存・文字検索・エディタへの直接転送・表示領域の可変・逆スクロール・容量設定・バックログリセット・バックログメモリ使用量表示・バックログ参照時に通信が可能) ★通信画面からのバックログスクロール。(バックログを開いて通信を行なっている最中に、通信画面上からバックログ画面をスクロールさせることができます。)★オリジナルエディタの搭載。(指定範囲直接通信・保存・文字列検索・文字列置き換え・指定行ジャンプ・部分コピー・エディタ領域の可変・エディタで編集中に通信が可能)★ヒストリ(UNDO)機能・編集機能・(11個までのヒストリー・1ラインエディタによる文字列の編集・登録)★通信中に子プロセスによるHumanコマンドの実行。(実行コマンドの事前登録が可能)★自動実行トレース表示機能。★ファイル内容表示。★ファイル一覧表示・選択。(ファイルソート・サーチ機能)★指定パス・ディレクトリのツリー表示機能。(パスの事前登録が可能)★ディレクトリー覧表示・選択。★ヘイズAT・CCITT・MNPモデム対応。★半角カタカナの平仮名変換表示。★ローカルエコー可能。★16進表示による受信文字列表示機能。★ブレーク信号送信時間設定機能。★画面表示色の設定変更可能。★232C割り込みインジケータ表示。★画面モードの変更可能(24KHz、31KHz)★カラムゲージ表示機能。★チャット用1ラインエディタ編集。★ファンクションキー(F1〜F20)・カーソル移動キーの開放によりユーザー設定可能。★ユーザーキーの設定(アルファベットA〜Zまでに文字列設定可能)★通信終了時のバックログ自動または指定保存機能。

「た〜みのる」ユーザーに差額交換サービス実施中!!
ユーザー登録をされていない方は早目に愛用者カードをお送り下さい。

好評発売中

コナミのパズルゲーム「キューブリック」のX68000の移植版

キューブランナー CUEB RUNNER

……あなたは、全面制覇できますか?……
レールのついた15枚のブロックを巧みに組み合わせてモアイの乗ったキューブを時間内に全部のレールを通過させれば面クリアです。とにかく、夢中になること間違いなし。じっくりパズルゲームで過ごしましょう。

ネメシス'90 NEMESIS

開発順調

¥8,800

「ネメシス」このネーミングに聞き覚えはありませんか？そうです!!あのグラディウス2(MSX版)の海外名なのです。これを、大幅にアレンジX68000用「グラディウス2」SPSオリジナルとして移植、開発中!! X68000のグラディウスシリーズがまた一つあなたのコレクションの中に増えますよ!!!

ワールドコート

近日発売

¥8,800

あの「ワールドコート」が、X68000版で新登場!!
ゲームセンターで手に汗を握ってプレイしたテニスの興奮をX68000で再現します。もちろん、ひょうきんQ3−C3・Q2−C2を忠実に再現して皆様のお相手をします。

SPS-NET

TEL (0245)46-1167(代)

Tri-P 好評運営中(4回線) / 一般回線(9回線) MNPクラス7

24時間運営(N81XN)
ゲストID(GUEST)
※GUESTアクセスは無料ですのでぜひ、一度試してください。

入会方法

登録料¥3,000(税別)
会費無料

下記の用紙に直接記入するか又は、コピーして記入し、72円切手同封の上、「SPS-NET係」までお送り下さい。届き次第、仮登録を行いID発行後SPS-NET専用の郵便振込み用紙ならびに運用の手引きをお送りいたします。それに従い、3ヶ月以内に登録料3,000円(税別)を御入金下さい。入金確認後正式会員として再登録します。

例◎パスワード＝SPS-NET (8文字まで大小文字の識別あり)
(　　　　)

◎本名＝大和大五郎(8文字まで)
(　　　　)

◎ペンネーム＝大ちゃん(4文字まで)
(　　　　)

◎年齢＝30(現在の年齢)
(　　　　)

◎電話＝0245-45-5777(市外局番から)
(　　　　)

◎職業＝株式会社エス・ピー・エス(16文字まで)
(　　　　)

◎住所＝福島市太平寺字町ノ内5 3(24文字まで)
(〒　　　　)

◎自己紹介＝SPS-NETをよろしく(24文字まで)
(　　　　)

◎システム構成＝X68000ACE-HD MD2400B(18文字まで)
(　　　　)

★Tri-P資料(必要・不要)
Tri-P資料不要の場合62円切手を同封してください。

好評発売中!

「棋太平」各種発売中
…………………………価格T4,500円5″FD **6,500円**
…………………………5″2HD 3.5″FD **7,000円**

「ザ・リターン・オブ・イシター」
……PC-8801以降PC-9801シリーズX1ターボFM77価格 **6,800円**

「た〜みのる」X68000…………価格 **12,800円**

「HOST PRO-68K9」…価格 **59,800円**

「HOST PRO-68K3」…価格 **39,800円**

「JETターボ・ターミナル」………価格 **9,800円**

(株)マイコンハウス SPS
〒960 福島市太平寺字町ノ内5-3 ☎(0245)45-5777
FAX(0245)45-1804(GII,GIII)

当社の製品は全国の有名テハート、ハソコンショッフてお求めになれます 尚、お求めになれない場合、郵便局にてお申し込みください ●口座番号 郡山5-12298 ●加入者名㈱エス・ピー・エス ●金額 代金に3%の消費税を加算した額●通信欄(裏面)ご希望ゲームソフト名、数量、代金合計、年齢、氏名、機種名、テープかディスクの種類(一週間以上かかりますので、お急ぎの方は現金書留をご利用ください その場合、おつりのいらないようにお願いします

■表示価格に消費税は含まれておりません。

多数の方からのご要望にお応えしてX1ターボ版ついに誕生！
品切れ店続出！宿題が楽になったと大人気！究極のずるかましソフト！

翻訳ヘルパーずるかまし

対応機種：X68000(5インチ2HD)、X1ターボ（5インチ2D）各2枚組 ¥5,980
開 発 者：(X68000版) 大阪市立大学マイコン研究会 山本賢一
(X1ターボ版) STUDIO ATTIC 水無月みるく
辞書作成：大阪市立大学マイコン研究会 山本博之
《内容・特長》英文翻訳ガイド、英和辞書、和英辞書、英単語暗記トレーニング、辞書ユーティリティからなる翻訳の友。辞書4800語付き。

太陽系を舞台としたわが国初のエコロジーシューティングゲーム

MEGA PRESSURE（メガ・プレッシャー）

対応機種：X68000(5インチ2HD) 2枚組 ¥5,980
開 発 者：関西学院大学電脳研究会 池田尚隆
《内容・特長》理想的なゲームバランス。バリエーションに富んだBGM。環境破壊に対する警告を含んだ問題作。

史上初、3Dで秒間描写30コマ以上を実現！
360度全方向への進行可能、究極の3Dドライビングゲーム

F. T. SCAN（エフ・ティ・スキャン）

対応機種：X68000(5インチ2HD)、X1/X1ターボ（5インチ2D） ¥5,980
開 発 者：Final Tear Z / Seafy・NAZ・Spark
《内容・特長》バトル、レース、鬼ごっこ各モードを備えた、F.T.SCANシステム搭載の3Dアクションゲーム。2人同時PLAY可能。通信による4人同時PLAYも可能。X1、X68000共サイバースティック対応。

開発中！ X68000用ソフト 各5,980円

ビジュアルシーンふんだん、涙と感動のストーリ展開！
全150面 時間制限無しの究極の熱中型パズルゲーム

HOP UP（ホップ アップ）

開 発 者：関西学院大学電脳研究会 池田尚隆・浅田真一・河野匡格

4系列5段階の対空パワーアップ！3種類射程3段切り替えの対地弾！
縦スクロールシューティングゲーム衝撃の超大作。

Task Force ALFARNE（タスクフォース・アルファーン）

開 発 者：Shillpheed Soft 野村恵・磯野友厚・小村俊平

究極美表現エキサイトX指定第2弾！
本格的ファンタジーアドベンチャーゲーム

AQUARIUS（アクエリアス）

原作・開発者：神戸大学情報統計部 赤坂賢洋
グラフィック：神戸大学情報統計部 細見格・中野博之
《内容》
魔法により、石に変えられ、洞窟に閉じ込められた恋人を助け出すため、オルフェウスという若者が旅するというストーリーで、グラフィックに究極の美しさを追求したファンタジーアドベンチャーゲーム。

郵送品貼付切手には、オール記念切手使用！

日コン連SOFT通信販売のご案内

現金書留、郵便振替(大阪5-4873 日コン連企画株式会社)、為替、定額小為替で、希望商品名、対応機種名、数量明記の上、お申し込みください。(送料はサービス。)
このうち、現金書留、定額小為替でお申し込みの場合には、例えば5,980円の商品の場合には、端数を切上げ6,000円分お送りいただいて結構です。この際のおつり20円は、商品発送時に同額の記念切手でお返しいたします。

日コン連発行のパソコン雑誌「C・able」の通信販売のお知らせ
ご希望の方は、切手360円分同封の上"C・able"希望と明記の上、お申しつけください。

好評発売中！ X68000用ソフト

歴史に残る！日コン連SOFT初のシューティングゲーム！
たった一人でプログラムからグラフィック、ミュージックまで担当！

D_RETURN

原作・開発者：神戸大学情報統計部 赤坂賢洋 ¥5,980
伝説のソフト！在庫譲少の為、お買い求めをお急ぎください。！

アドベンチャーゲームインタプリタ！

電脳作家 Ver2.0

原作・開発者：神戸大学情報統計部 村尾元 ¥5,980
プログラムが組めなくても、アドベンチャーゲームが作れます！
グラフィックツール、サンプルシナリオ付き。
史上初、自作シナリオの商品化チャンス付き。

電脳作家用グラフィック、ミュージックデータ集

電脳作家グラフィック&ミュージックライブラリー集

¥3,980
制 作 者：神戸大学情報統計部 細見格・赤坂賢洋
D_RETURNゲームミュージックが自作ソフトのBGMに使えます。

電脳作家シナリオ集①

EVIL EYE：作 三上潤一郎 ¥2,980
スターマンの伝説：作 川合一広

自作シナリオディスクを日コン連に送ると、もれなく商品化されます。
- シナリオ集1を購入されユーザー登録される方で、ユーザー登録時にモニター希望と書かれた人については、先着30名様に限り日コン連SOFTの新作モニターとなっていただきます。なお、すでにユーザー登録をお済ませの方は、官製ハガキにモニター希望とお書きの上、日コン連の方までお送りください。

お知らせ！

日本コンピュータクラブ連盟加盟団体募集中！

加盟費・会費不要。毎月、全国本部広報紙「つうてんかく通信」無料送付。

■日コン連では、以下のスタッフを求めています。
- 日コン連全国本部（難波）、関東本部（自由が丘）付けスタッフ
- 日コン連コンピュータウィルス研究所非常勤スタッフ
- パソコン雑誌「C・able」のライター及びエディター
- 日コン連陸上部、水泳部の部員

《お問い合わせは、下記まで。》
日コン連全国本部 06-644-6901(代)/日コン連関東本部 03-702-2891

●日コン連SOFTユーザー登録のお勧め
日コン連SOFTのユーザー登録をされますと、日コン連関連の様々な便宜がはかられます。ユーザー登録をされることをお勧めします。
日コン連コンピュータウイルス研究所には、連日、X68000ユーザーを中心として、多数のウイルスについての問い合わせや相談が寄せられています。特に、かなりの数のX68000に難波1号という名前のウイルスが入り込んでいるようです。
難波1号…1989年1月1日制作と、1月15日制作の2つのバージョンが存在している。症状は、1990年7月より月1回フロッピィディスクまたは、ハードディスクの内容を消去するというもの。本年6月までは、潜伏期間で直接の被害は出ないことになっているが、そのウイルスに感染しているマシンでは、一部のゲームが起動しなくなる等の被害が出ている。
日コン連コンピータウイルス研究所(大阪)では、日コン連SOFTのユーザー登録されている方については、ご自分のマシンにかかったウイルスの検出、復元のご相談に無料で応じています。また、日コン連コンピュータウイルス研究所開発のワクチンについては、その配布を登録ユーザーの方に優先的に行います。お気軽にご相談ください。
●「サークル日コン連」（日コン連加盟）会員（個人）募集中！
●コンピュータクラブ入会・新設斡旋
大学に入学してコンピュータクラブに入りたい、または、コンピュータクラブをつくりたいとお考えの方は、日コン連にご相談ください。クラブのご紹介や新設される際の機材の貸出等のサービスを行います。

日コン連SOFT保証

お客様のご都合により、同一種の新しいディスクとの交換を希望される場合には、そのディスクと360円分の切手をお送りください。折り返し、新しいディスクをお送りさせていただきます。

●問い合せ・申し込み先

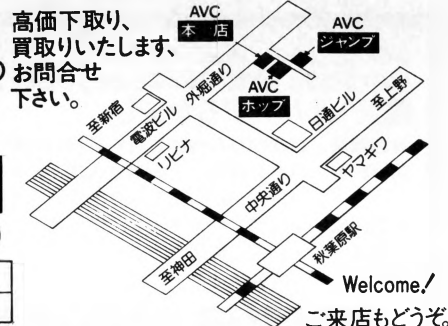
〒556 大阪市浪速区難波中2-4-3 村上ビル
TEL 06(644)6901(代)
日コン連企画株式会社または日本コンピュータクラブ連盟

秋葉原価格でローンができます
電気の街秋葉原で
25年
の信用!!

AVCフタバ

AUDIO VIDEO COMPUTER

☎03(253)7661

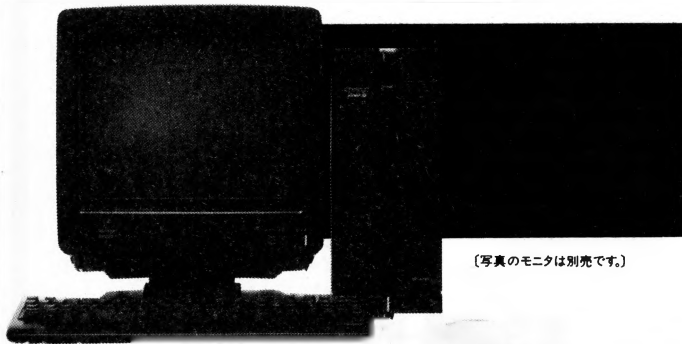
〒101 東京都千代田区外神田3-2-3 ☎03-253-7611(代)

今すぐ もよりの電話から	仙　台 022-264-3704	名古屋 052-452-3271	広　島 082-295-6873
札　幌 011-611-5104	新　潟 0252-75-4175	大　阪 06-311-3931	福　岡 092-481-2494

高価下取り、
買取りいたします、
お問合せ
下さい。

AVC 本店
AVC ジャンプ
AVC ホップ
至新宿
電波ビル
外堀通り
日通ビル
至上野
リビナ
ヤマギワ
中央通り
至神田
秋葉原駅
Welcome!
ご来店もどうぞ。

X68000の情報のすべて!(当店はX68000の認定代理店です。お気軽にご相談下さい)

X68000 待望の新しい仲間登場!!

X68000
PERSONAL WORKSTATION
EXPERT・EXPERT HD

〔写真のモニタは別売です。〕

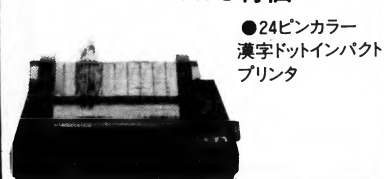
CZ-602C　標準価格¥356,000
CZ-612C　標準価格¥466,000
AVC特価

X68000
PERSONAL WORKSTATION
PRO・PRO HD

拡張I/Oスロットを4スロット標準装備、メインメモリ1MB、Human68K ver2.0搭載(CZ-652C)更に40MBのHDDを搭載(CZ-662C) 新しいX68Kの発見があるはずだ。
〔写真のモニタは別売です。〕

CZ-652C　標準価格¥298,000
CZ-662C　標準価格¥408,000
AVC特価

CZ-8PG2

標準価格¥160,000
⇒AVC特価

●24ピンカラー漢字ドットインパクトプリンタ

お勧めディスプレイコーナー　組合せは自由、価格はお気軽にご相談下さい。

CZ-604D
標準価格¥94,800
AVC特価
●0.31mmドットピッチ
●2モードオートスキャン
●ステレオスピーカ搭載
●チルト台同梱

CZ-612D
標準価格¥118,800
AVC特価
●0.31mmドットピッチ
●TVチューナ搭載
●3モードオートスキャン
●チルト台同梱

CZ-603D
標準価格¥84,800
AVC特価
●0.31mmドットピッチ
●TVチューナ無し
●3モードオートスキャン
●チルト台同梱

CU-21HD
標準価格¥148,000
AVC特価
●0.52mmドットピッチ
●21型ディスプレイ
●3モードオートスキャン
●ステレオスピーカ搭載

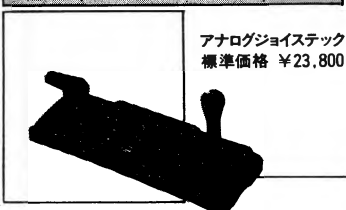
CZ-602D
標準価格¥99,800
AVC特価
●0.39mmドットピッチ
●TVチューナ搭載
●3モードオートスキャン
●チルト台同梱

CU-21CD
標準価格¥139,800
AVC特価
●0.52mmドットピッチ
●TVチューナ無し
●3モードオートスキャン
●チルト台取付不可

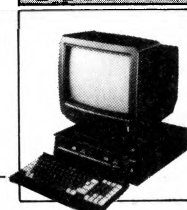
型　番	品　名	標準価格	販売価格
CZ-6TU	システムチューナー	¥ 33,100	AVCフタバ特価
BF-68PRO	CRTフィルター	¥ 19,800	AVCフタバ特価
CZ-8NS1	カラースキャナー	¥188,000	AVCフタバ特価
CZ-6BN1	スキャナー用パラレルボード	¥ 29,800	AVCフタバ特価
CZ-6VT1	カラーイメージユニット	¥ 69,800	AVCフタバ特価
CZ-8BV2	カラーイメージボード	¥ 39,800	AVCフタバ特価
CZ-8BR1	立体映像セット	¥ 29,800	AVCフタバ特価
CZ-8DT2	パーソナルテロッパ	¥ 44,800	AVCフタバ特価
CZ-8BS1	FM音源ボード	¥ 23,800	AVCフタバ特価
CZ-8NJ1	ジョイカード	¥ 1,700	AVCフタバ特価
CZ-8NM2A	マウス	¥ 6,800	AVCフタバ特価
CZ-8NM3	マウス・トラックボール	¥ 9,800	AVCフタバ特価
CZ-6SD1	システムラック	¥ 44,800	AVCフタバ特価
AN-S100	アンプ内蔵スピーカー	¥ 36,600	AVCフタバ特価
CZ-6EB1	拡張I/Oボックス	¥ 88,000	AVCフタバ特価

型　番	品　名	標準価格	販売価格
CZ-8PG1	24ピンカラープリンター(80桁)	¥130,000	AVCフタバ特価
CZ-8PK10	24ピンプリンター(136桁)	¥ 97,800	AVCフタバ特価
IO-735X	カラージェットプリンター	¥248,000	AVCフタバ特価
CZ-6BE1A	1M増設RAMボード	¥ 38,000	AVCフタバ特価
CZ-6BE2	2M増設RAMボード	¥ 79,800	AVCフタバ特価
CZ-6BE4	4M増設RAMボード	¥138,000	AVCフタバ特価
CZ-6BP1	数値演算プロセッサー	¥ 79,800	AVCフタバ特価
CZ-6BC1	FAXボード	¥ 79,800	AVCフタバ特価
CZ-6BM1	MIDIボード	¥ 26,800	AVCフタバ特価
CZ-6BU1	I/Oボード	¥ 39,800	AVCフタバ特価
CZ-6BL1	LANボード	¥268,000	AVCフタバ特価
CZ-243BS	サイバーノート	¥ 19,800	AVCフタバ特価
CZ-240BS	スティショナリー	¥ 14,800	AVCフタバ特価
CZ-223CS	通信ソフト	¥ 19,800	AVCフタバ特価
ゲームソフト		20% OFF	

型　番	品　名	標準価格	販売価格
CZ-8TM2	モデムユニット	¥ 49,800	AVCフタバ特価
CZ-252MS	Musicstudio	¥ 28,800	AVCフタバ特価
CZ-247MS	MUSIC(MID)	¥ 28,800	AVCフタバ特価
CZ-221HS	NEW Print Shop	¥ 19,800	AVCフタバ特価
CZ-228BS	TOP給与計算エキスパート	¥200,000	AVCフタバ特価
CZ-227BS	TOP財務会計	¥200,000	AVCフタバ特価
CZ-220BS	DATA	¥ 58,000	AVCフタバ特価
CZ-212BS	BUSINESS	¥ 68,000	AVCフタバ特価
CZ-219SS	OS-9	¥ 29,800	AVCフタバ特価
CZ-211LS	Ccompiler	¥ 39,800	AVCフタバ特価
CZ-234LS	AI-68K	¥188,000	AVCフタバ特価
CZ-620H	20MBハードディスク	¥178,000	AVCフタバ特価
CZ-64H	40MBハードディスク	¥120,000	AVCフタバ特価
LHD-34V	40MBハードディスク(ロジテック)	¥153,000	¥117,000
LHD-32V	20MBハードディスク(ロジテック)	¥128,000	¥ 98,000

CZ-8NJ2

アナログジョイステック
標準価格 ¥23,800

AVC特価¥???

X1turboZIII

X1ターボシリーズの独自の機能を全継承、VCCIゼロdB基準に適合させた。
CZ-888C…¥169,800
CZ-860D…¥ 99,800
合計………¥269,600
特価　???

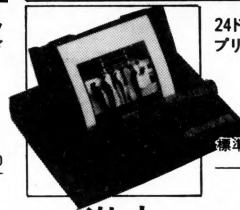
応談　価格はご相談に応じます、電話でお問い合せ下さい。

CZ-8PC4

48ドット熱転写プリンター。精密な文字、ハードコピーも可能。

CZ-8PC4……¥ 99,800

AVC特価¥???

CZ-8PC3

24ドット熱転写カラープリンター

標準価格……¥65,800

激安 価格はお電話にて

●但し消費税(3%)は別途請求させていただきます。●分割回数は3回~48回まで自由に選べます。

●セットの組合せは自由!広告に出ていない他の機種はお問合せ下さい。

●頭金なし(手軽な電話クレジット)●製品先取り(お支払いは約1~2ヶ月後から)●低金利クレジット(1回の支払いは2,700円以上で3~48回。ボーナス併用も可)●カレッジクレジット(保証人なし。但し満20歳以上の学生の方)●18歳未満の方(ご両親が代理購入者としてお申し込み下さい)●納期(通常の場合、当社に申込書が到着後1週間以内。特に人気のある商品で品薄の場合、少々納期が遅れることがありますので御了承下さい)●完全保証(すべてメーカー保証書付。アフターケア万全)●全国代引(お届けした者に、代金をお支払いいただく方法です。但し手数料1,000円)

AM10時からPM7時まで受付 日曜・祝日も営業

アイ・ツー EXE CLUB

全国初

新規ユーザー EXE会員 **店頭大集合**

★X68000ユーザーニーズに対応したハード・ソフト・ウエア・周辺機器は全て展示しています。
★新製品情報・ユーザー同士の情報交換ができる、メンバー様の憩いのスペースです。
★決算大特価セール期間中X68000・ディスプレイ・プリンター御購入の方は全国どこでも送料無料!!
★遠くでなかなかお越し頂けない方にも通販専用TELで専門スタッフ(X68 PRO STAFF)が親切丁寧にお答えします。
★X68000お買い上げの方、アイツーよりBigプレゼント。

X68000 オリジナルステッカー
X68000 フロッピータイトルシール
X68000 オリジナルテレフォンカード
X68000 バッグ
お好きなもの2点もれなくついてくる!!

★現在シャープX68000 EXE会員の方、おトモダチをご紹介下さい。ご購入成立時点でアイ・ツーとシャープよりステキなプレゼント進呈中!!
★アイ・ツーメンバーズ優待制度実施
アイ・ツーでX68000・及びソフトウエア周辺機器をお買上げ頂きましたユーザー様にはオリジナルメンバーズカードを送付致します。メンバーズの方には楽しいパソコンライフをおくれますように最善のフォローをアイ・ツーより提供します。

X68000プロショップ(専門店)ならではの企画です!!

決算展示品処分 大特価セット

たとえば

限定7台わずか!!

本体	
CZ-652C グレー・ブラック	C グレー・ブラック
モニター CZ-601D グレー・ブラック	モニター CZ-601D グレー・ブラック
¥272,000	**¥308,800**

あなたの支払う額はこれだけです!!

新品Setご相談に応じますドンドン☎下さい!!

★X68000新製品

Super-HD(CZ-623C)、EXPERT II(CZ-603C)
EXPERT II-HD(CZ-613C)、PRO II(CZ-653C)
PRO II-HD(CZ-663C)　ゾクゾク入荷!!

旧型製品も格安にて"御提供中"

X68000のことなら、なんでもご相談下さい。

- 全国をネットするアイ・ツーリサイクルシステム
- 独自の高価買取査定システム
アイ・ツーの買取指定金額は独自のシステムで計算しますので、高価で買取いたします。
- 東京、大阪店の店頭ではリサイクル商品を山づみ激安販売

FAX特集

SHARP FO-50	定価¥ 99,800	大特価にて展示中!
SHARP UX-10	定価¥128,000	
SHARP UX-20	定価¥168,000	

只今お買上の方、絶対必要なロール紙プレゼント中!!
全国どこでも送料無料。カタログ希望の方、下記の資料請求券をハガキに貼って御使用機種(CZ-とか)ご住所、ご氏名、TEL、生年月日をお書き添えのうえアイ・ツーEXE CLUBへご送付下さい。

X68000ユーザー様とっておきのグッズ!!

X68000オリジナルグッズをまだ持っていない君!
アイ・ツーからお届けしちゃいマス!

御礼

3月10日・11日のアイ・ツーサンクスフェアーに多数のご来店頂きまして誠にありがとうございました。第2回もただいま企画中です!乞うご期待!

4月は新入学フレッシュマンおめでとう

5月はゴールデンウィークバーゲンセール

■営業時間 AM11:00〜PM8:00

通販専用TEL.
06-634-0012
06-634-1198

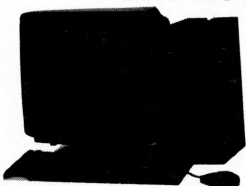
年中無休
Information & Interface
株式会社アイ・ツー
大阪店/〒542 大阪市中央区難波千日前15-18

信頼と実績のお店
BASIC HOUSE

X68000を御買上げの方にもれなく
下記X68000グッズのいずれか1つを
プレゼント!

A. PROSTAFF ジャンパー
B. X68000目覚し時計
C. ツタンカーメンZIPPO
D. ビジネスバッグ

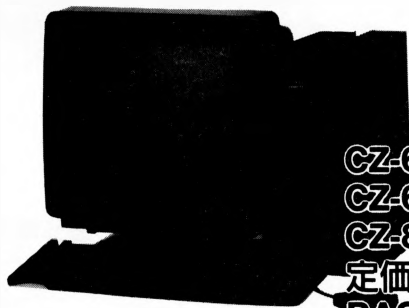
サポート万全! 我々にお任せください!

NEW X68000 SUPER-HD

○大容量80MB 3.5'HD内蔵
○SCSIインターフェイス標準装備
○疑似マルチタスク、マルチウィンドウを実現した"SX WINDOW"を搭載
○処理速度大幅向上(平均2倍)

特別価格にて予約受付中!

BASIC HOUSE超特価
(限定品)

CZ-602C-GY+CZ-612D-GY
¥3□□,000
CZ-652C-GY+CZ-612D-GY
¥3□□,000

NEW X68000 EXPERT II

CZ-613C
CZ-612D
CZ-8PC4
定価¥667,600
BASIC HOUSE特価

NEW X68000 PRO II

CZ-663C
CZ-605D
CZ-8PC3
定価¥609,800
BASIC HOUSE特価

AMIGA
CALL

周 辺 機 器

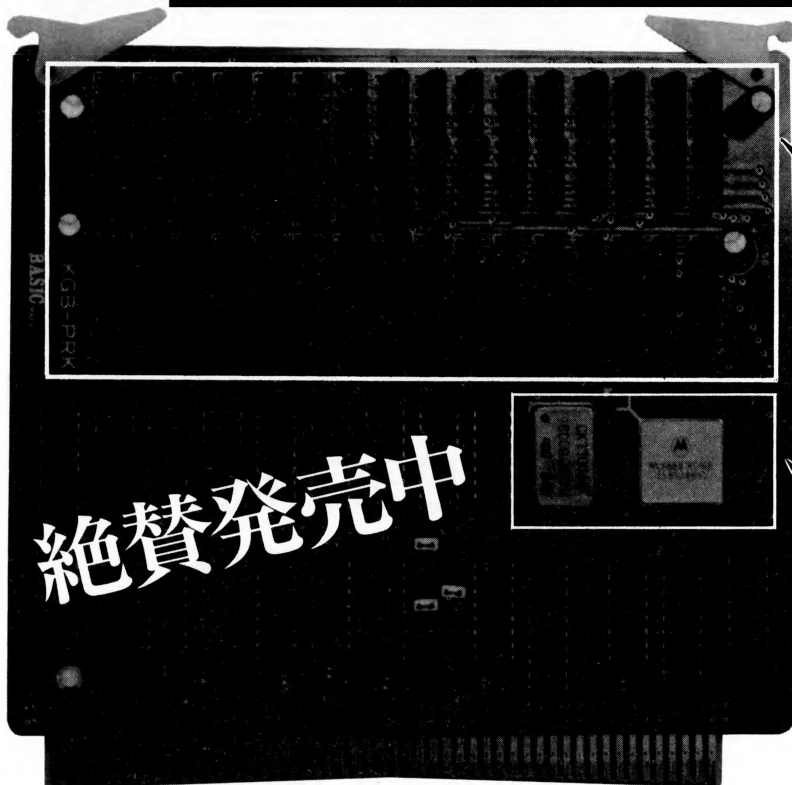
品名	価格	品名	価格	品名	価格
CZ-8PC3	BH特価	HXD042	¥128,000	SX-68M	¥ 19,800
CZ-8PC4	¥ 99,800	IT-X640	¥158,000	C Compiler PRO-68k	¥ 39,800
CR-3415CL	¥109,800	IT-X680	¥188,000	Mu-1	¥ 19,800
CR-3410CL	¥ 98,000	MD24FS5	BH特価	マジックパレット	¥ 19,800
VP-2050	BH特価	MD12FS	BH特価	Zs STAFF PRO-68k	¥ 58,000
CZ-8NS1	¥188,000	XE-1 PRO	BH特価	C-TRACE68	¥ 68,000
GT-6000	¥178,000	CYBER STICK	¥ 23,800	CARD PRO68k	¥ 29,800
GT-4000	¥198,000	CZ-6BF1A	¥ 38,000	CZ-6EB1	¥ 88,000
HS-10R II	¥ 49,800	CZ-6BG1	¥ 59,800	CZ-8NT1	¥ 13,800
HXD040	¥118,000	CZ-6BM1	¥ 26,800	AN-S100	¥ 36,600

全国どこでも発送可　長期クレジットOK　送料全国均一¥1,000　宅配便にて即日配送

株式会社計測技研
本社営業部　マイコンショップ　通販部　宇都宮市竹林町503-1　TEL0286-22-9811　FAX0286-25-3970
大田原営業所　マイコンショップ　大田原市美原1-13-4　TEL0287-23-5352　FAX0286-23-5364
マイコンショップ BASIC HOUSE お申し込み・お問い合せは ☎0286-22-9811(代)

2枚のボードが1枚になった

KGB-X68PRK

※写真はKGB-X68PRK-14です

広大なメモリ空間を実現する最大4Mバイトの
高速増設メモリ

高速演算を約束してくれる
数値演算プロセッサ

- メモリアクセスノーウェイトによる高速アクセス
- CZ-6BE2/CZ-6BE4/CZ-6BP1との混在が可能です
- 複数枚のKGB-X68PRKの実装が可能です
- ジャンパの変更により任意のアドレス空間にメモリの配置が可能です
- ジャンパの変更により数値演算プロセッサの1枚目2枚目/未使用の選択が可能です
- 1M/2M/3Mメモリモデルは購入後にメモリをボード上に追加可能です
- 数値演算プロセッサにはデバイスドライバ(FLOAT3X)が付属します
- 数値演算プロセッサにはMC68882も使用できます

※CZ-602C/CZ-612C以外の機種ではCZ-6BE1/CZ-6BE1Aを実装している必要があります
※メモリアクセスノーウェイトのため拡張I/O BOXでは動作しません

製品価格一覧

製品	価格
KGB-X68PRK-01 (1Mメモリ/数値演算プロセッサ無し)	¥58,000
KGB-X68PRK-02 (2Mメモリ/数値演算プロセッサ無し)	¥74,000
KGB-X68PRK-03 (3Mメモリ/数値演算プロセッサ無し)	¥98,000
KGB-X68PRK-04 (4Mメモリ/数値演算プロセッサ無し)	¥122,000
KGB-X68PRK-11 (1Mメモリ/数値演算プロセッサ付き)	¥96,000
KGB-X68PRK-12 (2Mメモリ/数値演算プロセッサ付き)	¥112,000
KGB-X68PRK-13 (3Mメモリ/数値演算プロセッサ付き)	¥136,000
KGB-X68PRK-14 (4Mメモリ/数値演算プロセッサ付き)	¥160,000

購入後の増設費用

メモリ	
1Mバイト	¥24,000
2Mバイト	¥51,000
3Mバイト	¥76,000
数値演算プロセッサ MC68881RC16	¥38,000

充実のBASICHOUSEハードウェア&ソフトウェア

製品	価格
高速12BIT,16CH A/Dコンバータボード(KGB-AD12) X1	¥118,000
フォトアイソレーション16BITデジタル入出力ボード(KGB-PIO) X1	¥42,000
ハードディスクインターフェースボード(KGB-HDIF) X1	¥16,000
アイソレーション16BITデジタル入出力ボード(KGB-X68PIO)X68000	¥68,000
ハンディプリンタ&インターフェース(HANDYPRINTjack) X68000	¥24,800
高速12BIT,4CH D/Aコンバータボード(KGB-DA4) X1	¥98,000
汎用ローコストA/D&PIOボード(KGB-X1S) X1	¥19,800
高速12BIT,16CH A/Dコンバータ(KGB-X68ADC) X68000	¥128,000
64180CPUボードMach180(KGB-CPXB) X68000	¥98,000
ローコストMIDIインターフェース(MELODY BOX) X68000	¥16,800

ソフトウェア	品番	価格
BASIC拡張関数パッケージ	B6-6301	¥9,800
C言語ライブラリ	B6-6305	¥6,800
BASIC拡張関数パッケージC言語ライブラリ付	B6-6306	¥14,800
ディスクキャッシャー	B6-6304	¥6,800
Toys & Tools	B6-6307	¥6,800
アイコンエディタ	B6 -6303	¥4,800
CP/M68Kエミュレータ	B6 -6302	¥19,800

BASICHOUSE BBS TECOSYS NET開局 TEL 0286-27-1829 1200 2400ボーMNPクラス5 8ビット パリティ無し X制御無し〈変更〉なし(ゲストIDは無くなりました。)

全国どこでも発送可 長期クレジットOK 送料全国均一¥1,000 宅配便にて即日配送

株式会社計測技研
本社営業部/マイコンショップ/通販部 宇都宮市竹林町503-1 TEL0286 22 9811 FAX0286 25 3970
大田原営業所/マイコンショップ 大田原市美原1-13-4 TEL0287 23 5352 FAX0286 23 5364

マイコンショップ BASIC HOUSE お申し込み・お問い合せは ☎0286-22-9811(代)

パソコンプラザ

店頭セール実施中

'90オクトで始まるパソコンワールド

☎03-730-6271

●営業時間 AM 11:00～9:00/日曜・祭日PM7:00 電話一本で、ハイ即納

〒144 東京都大田区蒲田4-6-7 FAX 03-730-6273

全国通販

●定休日毎週火曜日 祭日の場合翌日になります。

オクトラクラククレジット

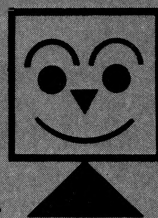
1回	2%	3回	2.5%	6回	3.5%	10回	5%	12回	5%	15回	7.5%
18回	9%	20回	10%	24回	11%	30回	14.5%	36回	15.5%	48回	20%

OCT-1 システム インフォメーション

- ▶全商品保証付(メーカー保証)
- ▶超低金利ハッピークレジット(1回～60回)頭金ナシOK!
- ▶ボーナス一括払いOK!ボーナス2回払いOK!!
- ▶配達日の指定OK!(万全なサポート体制)
- ▶商品の組合せ自由! オクトフリーダムシステム
- ▶店頭デモンストレーション実施中

オクト セレクテッドシステム

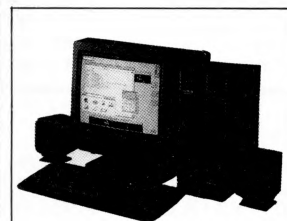
広告掲載商品以外の製品も取扱っております。

OCT-1 蒲田 OPEN

●平成2年、夏のボーナス一括払い(手数料ナシ) OKだよ～ん。超低金利 ハッピークレジットですゾ

EXPERT Ⅱ・PRO Ⅱ 新発売記念セール開催中!!

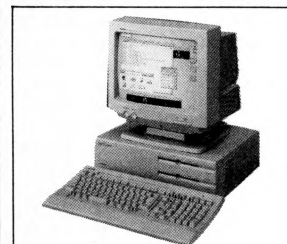
★下記セットでお買い上げの方にはプレゼント!! ●①MD-2HD 10枚 ②ジョイントカード(連射式) ③シリコンキーボードカバー ④アフターバーナー(¥9,200)

お好みのセットをお選び下さい。 送料無料

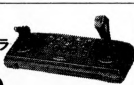
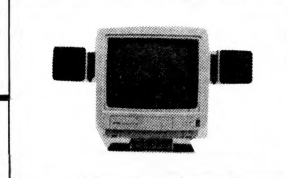
●SX-WINDOW搭載。
●40Mバイトハードディスク搭載

EXPERT Ⅱ・EXPERT Ⅱ-HD

●CZ-603C-BK/GY 定価¥338,000
●CZ-613C-BK/GY 定価¥448,000

現金特価!! 推選 お電話下さい。

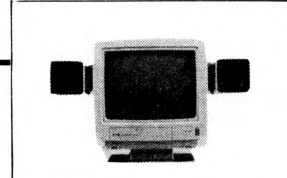
●SX-WINDOW搭載。
●拡張I/Oポート4スロット装備

PRO Ⅱ・PRO Ⅱ-HD

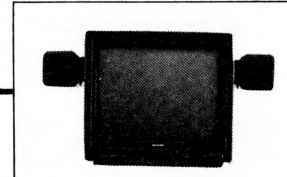
●CZ-653C-BK/GY 定価¥285,000
●CZ-663C-BK/GY 定価¥395,000

CZ-8NJ2
●インテリジェントコントローラ
定価¥23,800
超特価¥18,800

15型カラーディスプレイTV

CZ-605D-GY/BK 定価¥115,000

15型カラーディスプレイTV

CZ-613D-GY/BK 定価¥135,000

14型カラーディスプレー

CZ-604D-GY/BK 定価¥94,8000

21型カラーディスプレイ

CU-21HD 定価¥148,000

Ⓐ CZ-603C＋CZ-605D……………定価合計¥453,000▶**オクト大特価**
※オクトラクラククレジットをご利用下さい。
Ⓑ CZ-613C＋CZ-605D……………定価合計¥563,000▶**オクト大特価**
※夏のボーナス一括払い(手数料なし)!!
Ⓒ CZ-653C＋CZ-605D……………定価合計¥400,000▶**オクト大特価**
※配達日の指定OKだヨ～ン
Ⓓ CZ-663C＋CZ-605D……………定価合計¥510,000▶**オクト大特価**
※お店に遊びにおいでよ

Ⓔ CZ-603C＋CZ-613D……………定価合計¥473,000▶**オクト大特価**
※超低金利クレジットですので、ウフフですゾ!!
Ⓕ CZ-613C＋CZ-613D……………定価合計¥583,000▶**オクト大特価**
※クレジットは1回～60回まであるヨー
Ⓖ CZ-653C＋CZ-613D……………定価合計¥420,000▶**オクト大特価**
※店頭デモ実施中!!
Ⓗ CZ-663C＋CZ-613D……………定価合計¥530,000▶**オクト大特価**
※買って安心! TELくださーい。

Ⓘ CZ-603C＋CZ-604D……………定価合計¥429,800▶**オクト大特価**
※ガームもあるヨ、25% off!!から～もっと安くなるゾ。
Ⓙ CZ-613C＋CZ-604D……………定価合計¥542,000▶**オクト大特価**
※電話で値切っちゃえ! 親切だよ!!
Ⓚ CZ-653C＋CZ-604D……………定価合計¥379,800▶**オクト大特価**
※夏のボーナス一括払い(手数料なし)!!
Ⓛ CZ-663C＋CZ-604D……………定価合計¥489,800▶**オクト大特価**
※ハッピークレジットをご利用下さい。

Ⓜ CZ-603C＋CU-21HD……………定価合計¥486,000▶**オクト大特価**
※なにはなくても、クレジットがあればネ!!
Ⓝ CZ-613C＋CU-21HD……………定価合計¥596,000▶**オクト大特価**
※電話一本、ハイ即納!!
Ⓞ CZ-653C＋CU-21HD……………定価合計¥433,000▶**オクト大特価**
※ソフトのことならオマカセあれ!!
Ⓟ CZ-663C＋CU-21HD……………定価合計¥543,000▶**オクト大特価**
※今月もハ・ピ・プ・ペ・ポ!!

♡現金価格は、送料・消費税は別です。!!(送料¥2,000)

♡クレジット価格は、消費税込みですヨ。ご利用下さい!!

※クレジットの回数は1回～60回、ボーナス併用などありますのでお電話でお問合せ下さい。

■本体セット:送料無料 ●店頭デモ実施中…専門の係員が詳細にアドバイス致します。ぜひご来店下さい。

※上記料金には、消費税は含まれておりません。消費税が付加されますので、詳しくは、電話でお問合せ下さい。

■店頭にて、ゲームソフト25%OFF!!（税別）、超低金利 ハッピークレジットをご利用ください!!
■特に人気のある商品によっては、しばらくお待ち願うことがありますのでご了承下さい。

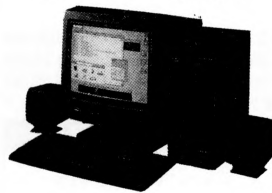
厳選された製品を、より安く、より早く、皆様のお手元に!!

広告掲載商品以外の製品も取扱っております。

チャンス！ X68000・SUPER（チタン）=6月発売!! 予約受付中!!

送料￥2,000

SX-WINDOW搭載。

他のディスプレイ①CZ-602D、②612D、③CZ-603D ④CU-21HDの組合せもございますのでお問い合せ下さい。

●ザ・ワークステーションと呼ぶにふさわしいスーパーな68000.!! 新登場!!
SUPER-HD。

※プレゼント！①MD-2HD10枚 ②アフターバーナー（￥9,200） ③ジョイカード（連射式） ④シリコンキーボード（￥2,800）

X68000 SUPER-HD

●CZ-623C-TN＋CZ-613D-TN
定価合計￥633,000…大特価!! TEL下さい。

※マウス・トラックボール付!! ディスプレイにはスピーカ2個、チルト台付!!

♡現品価格は、送料・消費税は別です。（送料￥2,000）
♡クレジット価格は、消費税込みですヨ。

※超低金利クレジットご利用下さい。1回〜60回払い、頭金ナシ！ボーナス1回払い、ボーナス2回払いOK！

オクト面白GOODS!!

アイテック （送料￥1,000）
X68000専用ハードディスク
アイテック
●X68000専用ハードディスク
◎IT-X640（定価￥158,000）
●40MB ●アクセスタイム28ms
特価￥103,000
◎IT-X680（定価￥198,000）
●80MB ●アクセスタイム20ms
特価￥134,000

限定

オクト特選 シャープ周辺機器 （送料￥1,000）

型名	商品	定価	特価
●CZ-6BE1	1BM増設RAMボード	￥35,000	特価￥26,500
●CZ-6BE1A	1MB増設RAMボード	￥38,000	特価￥28,800
●CZ-6BE2	2MB増設RAMボード	￥79,800	特価￥60,500
●CZ-6BE4	4MB増設RAMボード	￥138,000	特価￥104,800
●CZ-6BF1	増設用RS-232Cボード	￥49,800	特価￥38,500
●CZ-6BG1	GP-IBボード	￥59,800	特価￥45,000
●CZ-6BM1	MIDIボード	￥26,800	特価￥20,500
●CZ-6BN1	スキャナ用パラレルボード	￥29,800	特価￥22,800
●CZ-6BP1	数値演算プロセッサボード	￥79,800	特価￥60,500
●CZ-6BO1	ユニバーサルI/Oボード	￥39,800	特価￥30,500
●CZ-6EB1/BK	拡張I/Oボックス	￥88,000	特価￥66,800
●CZ-6VT1/BK	カラーイメージ・ユニット	￥69,800	特価￥53,000
●CZ-6BL1	LANボード	￥268,000	大特価
●CZ-8NM2A	マウス	￥68,800	特価￥5,300
●CZ-8NT1	マウストラックボール	￥98,800	特価￥7,500
●CZ-8NS1	カラーイメージスキャナ	￥188,000	大特価
●CZ-6BC1	FAXボード	￥79,800	特価￥60,500
●CZ-8TM2	モデムユニット	￥49,800	特価￥38,000
●CZ-64H	増設ハードディスク	￥120,000	大特価
●CZ-6TU GY/BK	RGBシステムチューナー	￥33,100	特価￥25,000
●BF-68PRO	高性能CRTフィルター	￥19,800	特価￥15,500
●SX-68M（システムサコム）	MIDIボード	￥19,800	特価￥15,000
●PIO-68BE1-A（I/O DATA）	1MB増設RAMボード	￥25,000	特価￥18,500
●PIO-6BE2-2M（I/O DATA）	2MB増設RAMボード	￥50,000	特価￥37,000
●PIO-6BE4-4M（I/O DATA）	3MB増設RAMボード	￥88,000	特価￥65,000

モデム・コーナー （送料￥1,000）

オムロン
●MD-1200AIII…特価￥14,800
●MD-24FS4……特価￥31,500
●MD-24FS5……特価￥34,800
●MD-24FP4……特価￥27,900

熱転写カラー漢字プリンター （ケーブル用紙付） 送料￥1,000

CZ-8PC4 ￥99,800

●48ドットサーマルヘッド
●B5〜B4まで
●ハガキ可能
●カラー対応

大特価 オクト推選 TEL下さい！

①CZ-8PC3（24ドット熱転写カラー漢字プリンター）
定価￥65,800……特価￥45,000
②CZ-8PK9（24ピン漢字プリンター80桁）
定価￥89,800……大特価!! TEL下さい。
③CZ-8PK10（24ピン漢字プリンター136桁）
定価￥97,800……大特価!! TEL下さい。
④CZ-8PG1（24ピンカラー漢字プリンター80桁）
定価￥130,000……大特価!! TEL下さい。
⑤CZ-8PG2（24ピンカラー漢字プリンター136桁）
定価￥160,000……大特価!! TEL下さい。
⑥IO-735X（カラーイメージジェット）
定価￥248,000……大特価!! TEL下さい。

パソコンラック 推奨 送料無料

①五段キャスター付

5段キャスター付
キーボードが収納できるから、手元でマウス操作がラクラクできる
棚板5段のマルチに活用できるディスク。
ウーン、こいつはデキル！
1325(H)×640(W)×700(D)
特価￥16,000

②四段キャスター付

4段キャスター付
どんなパソコンにもフレキシブルに対応！
使い易いデスクです。
1245(H)×614(W)×600(D)
特価￥12,000

X68000ソフト大セール実施中 ※ゲームソフトオール25%off

〈グラフィック〉●Z's STAFF PRO68K Ver.2.0
（シャフト）定価￥58,000
オクト特価￥40,500

〈データベース〉●KAMIKAZE
（サムシンググッド）定価￥68,000
オクト特価￥46,500

〈グラフィック〉●C-TRACE68
（キャスト）定価￥68,000
オクト特価￥51,000

〈C言語〉●C & Professional Pack
（マイクロウェアジャパン）定価￥58,000
オクト特価￥44,000

〈グラフィック〉●サイクロン エキスプレス
定価￥78,000
オクト特価￥58,000

●限定!!
◉サイクロン
限定特価￥25,000

※＋￥20,000で、サイクロン エキスプレスに交換できます!!

型名	商品	定価	特価
CZ-211LS	Ccompiler PRO-68K	￥39,800	￥28,800
CZ-212BS	BUSINESS PRO-68K	￥68,000	￥48,000
CZ-213MS	MUSIC PRO68K	￥18,800	￥13,500
CZ-214MS	SOUND PRO-68K	￥15,800	￥11,500
CZ-215MS	Sampling PRO-68K	￥17,800	￥12,800
CZ-219SS	OS-9/X68000	￥29,800	￥21,000
CZ-220BS	DATA PRO-68K	￥58,000	￥41,000
CZ-221HS	New Print Shop PRO-68K	￥19,800	￥14,300
CZ-223CS	Communication PRO-68K	￥19,800	￥14,300
CZ-224LS	THE 福袋 V2.0	￥9,980	￥7,500
CZ-226BS	CARD PRO-68K	￥29,800	￥21,300
CZ-241BS	システム手帳リフィル集	￥9,800	￥7,500
CZ-242BS	活用フォーム集	￥9,800	￥7,500
CZ-244SS	Homan 68K Ver.2.0	￥9,800	￥7,500
CZ-247MS	MUSIC PRO-68K〔MIDI〕	￥28,800	￥20,800
CZ-240BS	Stationery PRO-68K	￥14,800	￥11,500
CZ-243BS	CYBER NOTE PRO-68K	￥19,800	￥15,200
EW		￥38,000	￥29,800
G-68K		￥14,800	￥11,400
E-68K		￥19,800	￥15,300

店頭ゲームソフトオール25%off！ビジネスソフト 25%より特価中

●尚、送料として1ケ￥500、2ケ￥700、3ケ以上で￥1,000となります（税別）

★通信販売お申込みのご案内★ 〒144 東京都大田区蒲田4-6-7 TEL：03-730-6271

お申込みはお電話でお願いします。お客様の〈住所〉〈氏名〉〈電話番号〉及び〈商品名〉をお知らせ下さい。●入金確認後ただちに商品をご送付いたします。

現金一括払い
銀行振込：お近くの銀行より（電信扱い）にてお振込み下さい。
現金書留：封筒の中に住所・氏名・商品名をご記入の上当社までお送り下さい。

クレジット
専用お申込用紙をお送り致します。ので、必要事項をご記入、ご捺印の上ご返送下さい。手続きは簡単です。

オクト ラクラク クレジット表

1回	2%	3回	2.5%	6回	3.5%	10回	5%
12回	5%	15回	7.5%	18回	9%	20回	10%
24回	11%	30回	14.5%	36回	15.5%	48回	20%

振込先
富士銀行 久ヶ原支店 ㊟No.1824
三菱銀行 蒲田支店 ㊟No.0278691
株式会社 億人（オクト）

※掲載の価格は変動しますので、まずは、お電話にてご確認ください。
※連休のお知らせ＝4/17(火)、18(水)は連休です!!
※上記料金には、消費税は含まれておりません。消費税が付加されますので、詳しくは電話でお問合せ下さい。
※銀行振込、または、現金書留でご注文の際には、あらかじめ電話でご確認の上、お申し込み下さい。

■平成2年 夏のボーナス一括払いOK!!（7月末）手数料ナシ!!おトクです。ぜひ!!超低金利クレジットをご利用下さい。

X68000及びFM-TOWNS全国出張サポート

ワールド イン アオヤマには、X68000、TOWNS専門スタッフが常時待機。購入のアドバイスから新作ソフト情報まで、購入後のサービスが全く違います。会員のお客様には会員ダイアルもあわせてご利用下さい。
私共にて購入いただきましたX68000及びFM-TOWNSはどこでも出張サポートが受けられます。
シャープ X PRO SHOP・富士通 FM-SHOP会加入のプロショップです。

X68000

● 以前当社にてX68000及びX-1を御購入いただいたお客様に限り、CZ-8PC4（定価¥99,800）を大特価にてお届けいたします。会員の方は会員ダイアルにてCall／
● X68000をセットでお買い上げいただいたお客様に限り、アスキーターボステックを特価¥4,300、XE-1PROを特価¥6,700またCTRACEを特価¥47,800にてお届けいたします。御注文の際に合わせてお申し込み下さい。

X68000には、ブラックとオフィスグレーの2カラーがあります。

X68000PRO X68000 Aコース

- CZ-652-GY（本体）……¥298,000
- CZ-611D-GY（0.31ディスプレーテレビ）……¥134,000
- CZ-8PC3（24熱転写カラープリンター）……¥65,800
- 御希望ゲームソフト（人気ソフト上記よりお選び下さい）¥サービス

合計 ¥505,600 ➡ ¥314,900

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000PROII X68000 新Kコース

- CZ-653C（本体）……¥285,000
- CZ-604D（0.31ステレオスピーカ付ディスプレー）・¥94,800
- 御希望ゲームソフト（人気ソフト上記よりお選び下さい）¥サービス

合計 ¥379,800 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000PROII X68000 新Lコース

- CZ-653C（本体）……¥285,000
- CZ-605D（0.39ステレオスピーカ付テレビ）・¥115,000
- 住友3M5'2HDブランクディスケット ・・¥18,000
- 御希望ゲームソフト（人気ソフト上記よりお選び下さい）¥サービス

合計 ¥418,000 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000PROII X68000 新Bコース

- CZ-653C（本体）……¥285,000
- CZ-604D（0.31ステレオスピーカ付ディスプレー）・¥94,800
- CZ-8NJ2（インテリジェントコントローラー）・・・¥23,800
- ナイトアームズ……¥9,700
- サンダーブレード……¥9,500
- スーパーハングオン……¥7,800
- パックマニア……¥7,800
- ジェノサイド……¥8,800
- ヴァリスII……¥9,800
- 住友3M5 2HDブランクディスケット ・・¥18,000
- 御希望ゲームソフト……サービス

合計 ¥475,000 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000PROII X68000 新Eコース

- CZ-653C（本体）……¥285,000
- CZ-605D（0.39ステレオスピーカ付テレビ）・・¥115,000
- 上海II……¥6,800
- 倉庫番パーフェクト……¥6,800
- TETRIS……¥6,800
- 信長の野望戦国群雄伝……¥9,800
- スーパー大戦略68K……¥8,800
- 住友3M5'2HDブランクディスケット・・¥18,000
- 御希望ゲームソフト……サービス

合計 ¥457,000 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000PROII X68000 新O（オー）コース

- CZ-653C（本体）……¥285,000
- CU-21HD（21インチステレオスピーカー付ディスプレー）・¥148,000
- CZ-6TU（TVチューナー）……¥33,100
- CZ-8NJ2（インテリジェントコントローラー）・・・¥23,800
- スーパーハングオン……¥7,800
- サンダーブレード……¥9,500
- 住友3M5'2HDブランクディスケット ・・¥18,000
- 御希望ゲームソフト……サービス

合計 ¥525,200 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000EXPERTII X68000 新Dコース

- CZ-603C（本体）……¥338,000
- CZ-604（0.31ステレオスピーカ付ディスプレー）・¥94,800
- 住友3M5'2HDブランクディスケット……¥18,000
- 御希望ゲームソフト（人気ソフト上記よりお選び下さい）¥サービス

合計 ¥450,800 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000EXPERTII X68000 新V（ブイ）コース

- CZ-603C（本体）……¥338,000
- CZ-605D（0.39ステレオスピーカ付テレビ）・・¥115,000
- CZ-8NS1（カラーイメージスキャナ）……¥188,000
- Xstar（40MHDD）……¥118,000
- CZ-6BN1（スキャナ用ボード）……¥29,800
- CZ-6VT1（カラーイメージユニット）……¥69,800
- IO-735X（カラーインクジェットプリンター）・・¥248,000
- Z's staff pRo 68K ver2.0……¥58,000
- MZ-IC48（プリンターケーブル）……¥7,800
- 住友3M 5'2HDブランクディスケット・・¥6,800
- 御希望ゲームソフト……¥18,000
- ……サービス

合計 ¥1,197,200 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000EXPERTII X68000 新R（アール）コース

- CZ-603C（本体）……¥338,000
- CZ-605D（0.39ステレオスピーカ付テレビ）・・¥115,000
- リップスティックアドベンチャー ……¥6,800
- ヴァリスII……¥9,800
- V'BALL……¥7,900
- パワフル麻雀2……¥7,800
- 住友3M5'2HDブランクディスケット・・¥18,000
- 御希望ゲームソフト……サービス

合計 ¥503,300 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000EXPERTII X68000 新Hコース

- CZ-613C（本体：40MHDD付）……¥448,000
- 住友3M5'2HDブランクディスケット……¥18,000
- 御希望ゲームソフト（人気ソフト上記よりお選び下さい）¥サービス

合計 ¥466,000 ➡ 現金特価

安すぎて表示できません。クレジットでもお申し込み出来ます。

X68000 HDDモデル基本セット X68000 新X（エックス）コース

- CZ-613C（本体：40MHDD付）……¥448,000
- CZ-604D-BK（0.31ステレオスピーカ付ディスプレー）¥94,800
- 住友3M5'2HDブランクディスケット・・¥18,000
- 御希望ゲームソフト……¥サービス

合計 ¥560,800 ➡ 現金特価

通信セット〔ソフトX Talk-68K（¥12,800）＋モデム MD12FS1200ボモデム（¥21,000）〕➡¥27,300　NEW Print Shop（¥19,800）＋グラフィックライブラリーVOL.2（¥8,800）➡¥21,800
X68000接続電子手帳セット〔ケーブルCE-200L（¥2,500）＋サイバーノート68K（¥19,800）＋電子手帳PA-8500（¥28,000）〕➡¥37,600

X68000をはじめソフト&周辺機器類は、当社池袋店・札幌店・旭川店・千葉店にて実演中です。各店X68000コーナーが常設されております。

X68000ソフト&周辺機器

品名	価格	品名	価格	品名	価格
Kamkaze	¥68,000 ➡ 現金特価	Communication PRO68K	¥19,800 ➡ 現金特価	ユニバーサル1/0ボード	¥39,800 ➡ 現金特価
サウンドPRO 68K	¥15,800 ➡ 現金特価	インテリジェントコントローラー	¥23,800 ➡ ¥18,900	MT-32（ローランドデジタルシンセサイザ）	¥64,000 ➡ ¥55,000
Z's STAFF PRO68X	¥58,800 ➡ ¥40,800	トラックボール	¥13,800 ➡ ¥12,000	RS232Cボード	¥49,800 ➡ 現金特価
C compiler PRO68K	¥39,800 ➡ 現金特価	MUSIC PRO MIDI	¥28,800 ➡ 現金特価	数値演算プロセッサー	¥79,800 ➡ 現金特価
ミュージックPRO68K	¥18,800 ➡ 現金特価	MIDIボード	¥26,800 ➡ 現金特価	FAXボード	¥79,800 ➡ ¥63,000
BUSINESS PRO68K	¥68,000 ➡ 現金特価	ミュージックスタジオPRO	¥25,800 ➡ 現金特価	CZ-611D-GY	¥134,000 ➡ ¥74,800
OS-9/X68000	¥29,800 ➡ 現金特価	カラーイメージユニット	¥69,800 ➡ 現金特価	CZ-613D	¥135,000 ➡ 現金特価
C-TRACE	¥68,000 ➡ ¥47,800	1MB RAMボード	¥38,000 ➡ 現金特価	カラーイメージスキャナ	¥188,000 ➡ 現金特価
DATA PRO68K	¥58,000 ➡ 現金特価	2MB RAMボード	¥79,800 ➡ 現金特価	たーみのる（通信ソフト）	¥12,800 ➡ 現金特価
CARD PRO68K	¥29,800 ➡ 現金特価	4MB RAMボード	¥138,000 ➡ 現金特価	40MBハードディスクXstar	¥118,000 ➡ ¥94,400
Sampling PRO68K	¥17,800 ➡ 現金特価	拡張1/0ボックス	¥88,000 ➡ 現金特価	MD12FS（1200ボモデム）	¥21,000 ➡ 現金特価
NEW Printshop PRO68K	¥19,800 ➡ 現金特価	GP-1Bボード	¥59,800 ➡ 現金特価	MD24FP4（2400ボモデム）	¥39,800 ➡ 現金特価

X68000シリーズ&X-1シリーズ周辺機器

型番	品名	価格	型番	品名	価格
CZ-6PV1	カラービデオプリンター	¥198,000 ➡ 現金特価	CZ-8BV2	カラーイメージボード	¥39,800 ➡ ¥32,800
AN-S100	アンプ内蔵スピーカ〔ステレオ〕	¥36,600 ➡ ¥29,800	CZ-8BS1	ステレオタイプFM音源カード	¥23,800 ➡ 現金特価
BF-68PRO	高性能CRTフィルター	¥19,800 ➡ ¥16,800	CZ-8PG1	24ドットカラー漢字プリンター	¥130,000 ➡ 現金特価
ジョイスティック	アスキーターボステック	¥6,800 ➡ ¥5,440	CZ-8PG2	24ドットカラー漢字15インチプリンター	¥160,000 ➡ 現金特価
X-1/X68000	ジョイカード（延長コード付）	¥3,200 ➡ ¥2,900	VP-1350X-68000	24ドット15インチ漢字プリンター〔ケーブル付〕	¥103,600 ➡ ¥72,000

下記周辺機器は現金特価をお電話にてお問い合せ下さい。本体と合せてお申込みの場合は、クレジット及び代金引換にてお承ります。

組合せ自由	激安金利にキャンパスクレジット	ゆっくり、お支払いは8ヵ月先から
各コース以外の組合せもコースをベースに周辺を合せたセット……お支払いだって御希望のパターンをお組みいたします。さあ、ご相談もお見積りも受注センターもしくは各店へお気軽に。	手続きカンタン、大学生の為の超低金利クレジット。20歳以上の学生の方は原則として保証人様には連絡いたしません。	クレジット業界最低の金利を有効に使って、支払いは最長8ヵ月後から始まるクレジットでも。

全国出張サポート

私共にてご購入いただいたX68000は全国出張サポートがうけられます。

★この誌面に掲載の商品の価格には消費税は含まれておりません。

東京都豊島区東池袋1-27-12 明治生命池袋ビル 〒170
池袋店 東京都豊島区東池袋1-28-6 〒170
福岡ショールーム 福岡市中央区赤坂1-11-13 〒810
ソフト店 東京都豊島区池袋パールシティ 〒170

SHARP

★CU-21HD〔ステレオスピーカ付21インチディスプレー〕…￥148,000➡現金大特価
★CZ-604D〔ステレオスピーカ付603Dディスプレー〕…￥93,000➡現金大特価

X68000PRO II X68000

品名	価格
CZ-653C〔本体〕	￥285,000
CU-21HD-BK(ステレオスピーカー付21インチディスプレー)	￥148,000
AN-8TU(TVチューナー)	￥33,100
住友3M5'2HDブランクディスケット	￥18,000
御希望ゲームソフト(人気ソフト上記より御選び下さい)	￥サービス
合計 ￥484,100 ➡ 現金特価	安すぎて表示できません。

クレジットでもお申し込み出来ます。

X68000PRO MIDIセット X68000

品名	価格
CZ-653C〔本体〕	￥285,000
CZ-604D〔0.31ステレオスピーカ付ディスプレー〕	￥94,800
CZ-6BM1(MIDIボード)	￥26,800
CM-32P(ローランドMIDI音源)	￥72,000
MA-12C X2(ローランドスピーカー)	￥28,000
Music PRO68K	￥28,800
Sound PRO68K	￥15,800
Music Studio	￥25,800
CN-20(ローランドエントリーパット)	￥22,000
CF-10(ローランドデジタルフェーダー)	￥22,000
銀河英雄伝説(MIDI対応)	￥8,800
住友3M5'2HDブランクディスケット	￥18,000
御希望ゲームソフト	サービス
合計 ￥647,800 ➡ 現金特価	

分割	頭金	ボーナス
￥10,600×36回	ボ¥30,000	頭なし
￥12,200×48回	ボなし	頭なし

X68000EXPERT II X68000

品名	価格
CZ-603C〔本体〕	￥338,000
CZ-605D(0.39ステレオスピーカ付テレビ)	￥115,000
住友3M5'2HDブランクディスケット	￥18,000
御希望ゲームソフト(人気ソフト上記より御選び下さい)	￥サービス
合計 ￥471,000 ➡ 現金特価	安すぎて表示できません。

クレジットでもお申し込み出来ます。

X68000EXPERT II X68000

品名	価格
CZ-603C〔本体〕	￥338,000
CZ-605D〔0.39ステレオスピーカ付テレビ〕	￥115,000
CZ-8PC3(24熱転写カラープリンター)	￥65,800
Z's staff PRO 68K Ver. 2.0.	￥58,000
GT-1000(スキャナー、ケーブル付)	￥87,300
NewプリントSHOP(CZ-221HS)	￥19,800
グラフィックライブラリVol 2(お正月用ソフト)	￥8,800
合計 ￥692,700 ➡ 現金特価	安すぎて表示できません。

クレジットでもお申し込み出来ます。

X68000 NEWビジネスセット X68000

品名	価格
CZ-613C〔本体:40MHDD付〕	￥448,000
CZ-613D〔0.31ステレオスピーカ付テレビ〕	￥135,000
Stationery PRO 68K(電子手帳データ管理ソフト)	￥14,800
PA-8600(電子手帳)	￥28,000
CE-200L(電子手帳68K接続ケーブル)	￥2,500
Communication PRO68K(通信ソフト)	￥19,800
New Print Shop PRO 68K(印刷ユーティリティー)	￥19,800
CZ-8PC3(24カラー熱転写プリンタ)	￥65,800
MD24FS4(オムロン2400bpsモデム)	￥39,800
黒色インクリボンパック(15個入り)	￥3,000
カラーインクリボンパック(15個入り)	￥4,000
住友3M5'2HDブランクディスケット	￥18,000
御希望ゲームソフト	￥サービス
合計 ￥798,500 ➡ 現金特価	

分割	ボーナス	頭金
￥8,800×36回	ボ¥20,000	頭¥200,000
￥10,300×48回	ボ¥28,000	頭なし

X68000プロフェッショナルホビーセット X68000

品名	価格
CZ-613C〔本体:40MHDD付〕	￥448,000
CZ-604D-BK(0.31ステレオスピーカ付ディスプレー)	￥94,800
CZ-8NJ2(インテリジェントコントローラー)	￥23,800
サンダーブレード(立体シューティングソフト)	￥9,500
テラッツオ(スプライト・エディター)	￥19,400
住友3M5'2HDブランクディスケット	￥18,000
御希望ゲームソフト	￥サービス
合計 ￥613,500 ➡ 現金特価	

分割	ボーナス	頭金
￥9,800×60回	ボなし	頭なし
￥10,100×36回	ボ¥30,000	頭なし

X68000お買上げのお客様へ

上記コースで御希望ソフトは「ニュージーランドストーリー」「沙羅曼蛇」「ツインビー」「フルスロットル」「パックマニア」「ビーチバレー」「アルカノイド」「熱血高校ドッチボール」のうちいずれかからお選び下さい。

今月の限定お買得品

AN-8TU

RGBシステムチューナー対象ディスプレイ
アナログRGB入力対応(15P)/200ライン対応のもの
KD863S、862、CU-14AD、BD、ED、603D
KD854・852には使用出来ません。

定価合計 ￥33,100 ➡ 安すぎて表示できません

分割	ボーナス	頭金
￥5,000× 6回	ボなし	頭なし
￥9,900× 3回	ボなし	頭なし

CZ-8PC4

48ドット熱転写カラープリンター

定価合計 ￥99,800 ➡ 現金特価

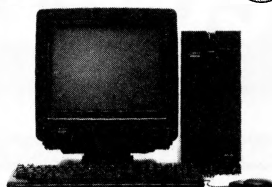
分割	ボーナス	頭金
￥3,800×24回	ボなし	頭なし
￥7,100×12回	ボなし	頭なし

SHARP CZ-8PC3

￥45,800

24ドット熱転写カラープリンター

定価合計 ￥69,800 ➡ ￥45,800

SHARP X68000 Sコース

￥258,000

品名	価格
CZ-652C〔本体〕	￥298,000
CZ-600D(ディスプレー)	￥139,800
合計 ￥437,800 ➡	￥258,000

SHARP X68000 Tコース

￥322,000

品名	価格
CZ-602C(X68000本体)	￥356,000
CZ-611D(0.3ドットカラーディスプレーテレビ)	￥134,000
合計 ￥490,000 ➡	￥322,000

SHARP X68000 限定コース Qコース

品名	価格
CZ-652CGY〔本体〕	￥298,000
CZ-611DGY(0.31カラーディスプレー)	￥134,000
合計 ￥432,000 ➡	￥261,000

クレジットでもお申し込みいただけます。

FMPR-204B FM TOWNS

FM-OASYS〔日本語ワープロ〕プリンターセット

カラー漢字熱転写プリンター

品名	価格
FMPR-204B	￥80,000
接続用ケーブル	￥6,800
FM-OASYS V1.0	￥55,000

〔FDD版高機能日本語ワープロソフト〕

合計 ￥141,800 ➡ 現金大特価

分割	ボーナス	頭金
￥3,900×36回	ボなし	頭なし
￥5,600×24回	ボなし	頭なし

FMPR-40T FM TOWNS

FM-OASYS〔日本語ワープロ〕プリンターセット

TOWNSカラー24ドット15インチ漢字プリンター新第1、2水準搭載、漢字80字/秒〔カラーユニットオプション〕FM-PR-354G同型プリンター

品名	価格
FMPR-40T(REM15インチ24漢字プリンター)	￥120,000
接続用ケーブル	￥6,800
FM-OASYS V1.0	￥55,000

〔FDD版高機能日本語ワープロソフト〕

合計 ￥181,800 ➡ 安すぎて表示できません

分割	ボーナス	頭金
￥5,100×36回	ボなし	頭なし
￥7,300×24回	ボなし	頭なし

FUJITSU FM TOWNS Eコース 一太郎ワープロセット

￥379,800

品名	価格
FM-TOWNS2(本体)	￥328,000
FMT-KB101(キーボード)	￥20,000
FMT-DP531(0.38ディスプレー)	￥89,800
TOWNSシステムソフト(OSver1)	￥20,000
TOWNSシステムソフト(MS-DOS)	￥18,000
My Fair Lady(英会話ソフト)	￥28,000
一太郎(ver3)(ジャストシステムワープロ)	￥68,000
合計 ￥571,800 ➡	￥379,800

クレジットでもお申込み出来ます。

X68000 1200ボーモデム電話付(EPSON SR-120PH 定価￥44,800➡特価￥23,000)

■各店X68000コーナー：TOWNSコーナーを常設しております。

■他店とは比較のできない厳選された商品です。

「T・ZONE」はおかげさまで満3才。これからもよろしくお願いいたします。

「オープン3周年記念謝恩セール」開催

4月25日(水)〜5月31日(木)

日頃のご愛顧に、感謝の気持ちをこめてとびっきりの大奉仕！
パソコン・ワープロ・無線機・あれやこれやが大特価！

ADO・TOYOMURA T・ZONE ティー・ゾーン

Micom Zone

2F 〒101 東京都千代田区外神田4-4-1 ☎257-2650

海外でも使える
「T・ZONE CLUB」
カード会員募集中!!
「オリエント」「UC」「マスター」カードが1つになった。
「ボーナス一括払い」OK！「通信販売」も
お手軽にご利用頂けます。そのほか、便利でお得な特典がいっぱい！ 今がチャンス!!
詳しくは、店頭にてどうぞ!!

X68000をトータルサポート
T・ZONE 2F
SHARP Authorized………

X68000 PRO SHOP

OS-9/X68000

- □OS9/68000 (SHARP) ¥29,800
- □C&PRO PACK (マイクロウェア) ¥58,000
- □MW-BASIC (マイクロウェア) ¥60,000
- □BTree09 (ARK) ¥36,000
  MW-BASIC用のISAM用B-Treeパッケージです。応用例として住所録と販売管理プログラムが付属。全ソースコード付です。(このソフトを動かすためにはMW-BASICが必要です。)
- □UD-CACHE (ARK) ¥16,000
  すべてのRBFデバイスに対応するキャッシュです。
- □FBU (ARK) ¥38,000
  ハード・ディスクバックアップユーティリティーです。巨大ファイルを分割バックアップしたり、日付管理を行なったバックアップもOK。
- □VSED (FORKS) ¥28,000
  OS9/68000で唯一オートバッファリングをサポートしたスクリーンエディタです。
- □Src Dbg 発売時期未だ不明

ニューリリースソフトウェア

CSG IMS V2.0
Clearbrook Software Group Information Management System
㈱星光電子 ¥118,000

あのCSG-IMSがついにX68000にリリースされました。高度の処理に対応可能な言語型リレーショナルデータベースです。フォーマッタを利用して簡単にシステム設計を行えます。OS9の特長を活かして、リアルタイム、マルチタスク、マルチユーザが必要なアプリケーションを構築することが可能です。もちろん、Cやアセンブラのモジュールを呼び出すことも可能、OAはもちろんFAにも対応できるパワフルツールです。

OS9-SHL FORKS
Super Shell for OS9/68K ¥12,800

お待ちかね、OS9-SHLがX68000に対応しました。
- ■プロシージャファイルで引数、リダイレクト、パイプが使用可能。
- ■環境変数を数値または文字列として演算可、制御文中で使用できます。
- ■豊富な制御文。While, wend, loopout, continue, if, else……
- ■ヒストリやエイリアスをサポート、マクロコマンドもOK。
- ■カーソルキーに対応。

Oh! FMコーナー 番外

FMシリーズ用OS9に新ソフト登場！

日本ソフトバンク
DB-09 (FM-7, 77, AV, 11) ¥18,252！
OS9上で走るリレーショナルデータベースマネージャーです。問い合わせ形式で取扱い簡単。なんとCによる全ソース付。

IMAGE and TEXT'S Inc.
Plot it！ (FM-11) ¥16,500
OS9上で走るプリント基板パターン設計用CAD。なんとVTerm 25にも対応。

CZ-8PC3 カラー熱転写プリンタ

X68Kのユーザーのプリンタ
所有率は意外に低い?!
このチャンスに1台考えてみてワ？

標準価格¥65,800⇒¥39,800

CZ-620H SHARP純正 20MBHD

嗚呼！ 遂に涙涙の¥49,800
標準価格¥178,000がついに！
これで本当におしまいです。

T・ZONE正社員・長期アルバイト募集中！
☆お問い合わせは総務課鈴木まで(TEL 03-257-2630)

T・ZONE
8F オフィス
7F 工学書 輸入
6F 衛星放送
5F HAM トランシーバー
4F トレードゾーン
3F IBMゾーン
2F パソコンソフトウェア
1F パーソナルOA
B1 Macintosh!

T・ZONE
営業時間：AM10:30〜PM7:00

下記T・ZONE各店でも扱っています。
宇都宮店：☎0286(63)4949 川口店：☎0482(68)7826 静岡店：☎0542(83)1331 横浜店：☎045(641)7741
大宮店：☎048(652)1831 東ラジ店：☎03(257)2694 パーツショップ：☎03(257)2655

●マイコン通販利用の方へ：現金書留て送金される際は、住所、氏名、TEL番号、希望商品名(詳しく)を明記して下さい 振込を御希望の方は下記銀行へお願いします
尚、いずれも予めTELにて、御予約・送料確認の上御送金下さい (振込口座 埼玉銀行 秋葉原支店 当座2705 ㈱亜土電子工業)

☆この広告の提示価格には、消費税は含まれておりません。

クリエイト特典

- ●全商品完全保証書付(メーカー保証)
- ●全国無料配達(一部離島の方は有料になります)
- ●配達日の指定OK(日曜・祭日にかかわらずお客様のご都合にあわせて配達します)
- ●どんな商品の組合せも自由自在(ご予算、用途に応じ自由自在にシステムアップできます)
- ●中古パソコン高額下取り(今お使いのパソコンをわずかな差額でグレードアップ)
- ●お支払い方法自由(低金利の均等払い、ボーナス一括払いもご利用ください)

営業時間(定休日▶渋谷店:日曜・祭日/横浜店:水曜)
AM10:00~PM7:00

当社はX68000の販売認定店です。どんなことでも安心してご相談ください。

X68000特別 新入学フェア開催中!!

X68000 NEW PRO II

- ●CZ-653C(本体)……￥285,000
- ●CZ-603D(カラーディスプレイ)……￥ 84,800
- ●お好きなゲームソフト1本……￥ 7,800
- ■定価合計……￥377,600

クリエイト特価

均等払い	￥ 7,680×48回	￥ 9,890×36回	￥14,370×24回
ボーナス	なし	なし	なし

X68000 NEW EXPERT II

- ●CZ-603C(本体)……￥338,000
- ●CZ-613D(カラーディスプレイテレビ)……￥ 99,800
- ●CZ-8NJ2……￥ 23,800
- ●お好きなゲームソフト1本……￥ 9,800
- ■定価合計……￥506,600

クリエイト特価

均等払い	￥ 9,970×48回	￥12,840×36回	￥18,660×24回
ボーナス	なし	なし	なし

X68000 EXPERT II HD

- ●CZ-613C(本体)……￥448,000
- ●CZ-604D(カラーディスプレイ)……￥ 94,800
- ●お好きなゲームソフト1本……￥ 9,800
- ■定価合計……￥552,600

クリエイト特価

均等払い	￥ 5,920×48回	￥ 7,400×36回	￥12,100×24回
ボーナス	￥30,000× 8回	￥40,000× 6回	￥50,000× 4回

X68000 SUPER HD

- ●CZ-623C-TN(本体・キーボード・マウス)……￥498,000
- ●CZ-613D-TN(カラーディスプレイ)……￥135,000
- ●CZ-6BP1……￥ 79,800
- ■定価合計……￥712,800

クリエイト特価

均等払い	￥ 7,320×48回	￥10,100×36回	￥13,450×24回
ボーナス	￥42,000× 8回	￥50,000× 6回	￥80,000× 4回

※本広告に掲載の全商品の価格について消費税は含まれておりません。

X68000シリーズ用 周辺機器・ソフト オール超特価!!

型番	品名	定価
CZ-6VT1	カラーイメージユニット	￥ 69,800
CZ-8NS1	カラーイメージスキャナ	￥188,000
CZ-6BE1A	IMB増設RAMボード	￥ 38,000
CZ-6BE2	2MB増設RAMボード	￥ 79,800
CZ-6BE4	4MB増設RAMボード	￥138,000
CZ-8NM3	マウス・トラックボール	￥ 9,800
BF-68PRO	高性能CRTフィルター	￥ 19,800
CZ-6BP1	数値演算プロセッサ・ボード	￥ 79,800
CZ-8NT1	トラックボール	￥ 13,800
CZ-6BM1	MIDIボード	￥ 26,800
AN-S100	アンプ内蔵スピーカースシステム	￥ 36,600
CZ-8NJ2	アナログスティック	￥ 23,800
CZ-603D	ドットピッチ0.31mm14型高解像度	￥ 84,800
CZ-6TU	パソコンチューナ	￥ 33,100

ソフト名	品名	定価
MUSIC PRO	MIDI版	￥ 28,800
MUSIC PRO-68K	マウスを使った楽譜ワープロ	￥ 18,800
SOUND PRO-68K	サウンドエディタ	￥ 15,800
Sampling PRO-68K	AD PCMサンプリングエディタ	￥ 17,800
Musicstudio PRO-68K V.1.1	MIDIマルチレコーディングソフト	￥ 28,800
NEW Print Shop PRO-68K	ポップアートツール	￥ 19,800
Communication PRO-68K	高機能通信ソフト	￥ 19,800
OS-9/X68000	マルチタスクオペレーティングシステム	￥ 29,800
PRO-68K	サイバーノート	￥ 19,800
PRO-68K	ステーショナリー	￥ 14,800
DATA PRO-68K	コマンド型リレーショナルデータベース	￥ 58,000
CARD PRO-68K	カード型リレーショナルデータベース	￥ 29,800
Ccompiler PRO-68K	ソフト開発セット	￥ 39,800
Human 68K Ver2.0	開発ツールセット	￥ 9,800

▲上記以外ビジネスソフト、最新ゲームソフト豊富に在庫あります。※送料はご注文の際お問合せください。 ●超特価販売中!

★この表以外の組合せ、お支払い方法もご自由にできます。
★X1シリーズ用、X68000シリーズ用各社ハードディスク/プリンタ等の周辺機器を大特価にて販売しております。
電話にてお問合せください。

●渋谷店

●横浜店

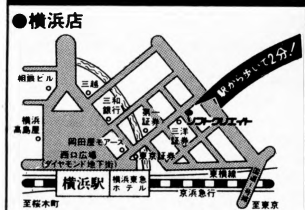
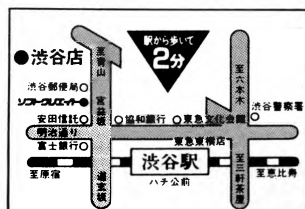
オール15%~20%OFF

総合お問合せ先☎03-486-6541(代)

パソコン専門ショップ

ソフトクリエイト 渋谷/横浜

●渋谷店☎03-486-6541(代)
〒150:東京都渋谷区渋谷1-12-7 三和渋谷ビル
振込銀行:三井銀行 渋谷宮益坂支店(普)No.5000340

●横浜店☎045-314-4777(代)
〒221:横浜市神奈川区鶴屋町2-12-8 第1建設ビル
振込銀行:三和銀行 横浜駅前支店(普)No.310852

お買い得　超特価セット（限定品）

（このセットに限り、送料＋消費税込）

X68000 EXPERT/PRO

New X68000新発売！

型番	定価
CZ-603C	¥338,000
CZ-613C	¥448,000
CZ-623C	¥498,000
CZ-653C	¥285,000
CZ-663C	¥395,000
〈ディスプレイ〉	
CZ-603D	¥84,800
CZ-604D	¥94,800
CZ-605D	¥115,000
CZ-613D	¥135,000

'90年5月末迄

CZ-602C（本体）	¥356,000
CZ-602D（ディスプレイ）	¥99,800
定価合計	¥455,800

ズバリ大特価！¥315,000

CZ-652C（本体）	¥298,000
CZ-602D（ディスプレイ）	¥99,800
定価合計	¥397,800

ズバリ大特価！¥270,000

●CZ-652、602単体でも大特価！お問い合わせください。

※代金は商品引換着払いでもOKです。

MZ2500下取り／MZ2500からMZ2861（定価¥328,000）に買い替え下取り後　特価¥165,000

FM TOWNS お買い得セット

1.TOWNS-1	¥338,000
2.FMT-ME(1M)	¥60,000
3.FMD-FD301	¥28,000
4.FMT-KB101	¥20,000
5.FMT-DP531	¥89,800
6.TOWNS-OS V1.1 L20	¥20,000
定価合計	¥555,800

大特価！¥285,000

ハガキもOK、New MZプリンタ

漢字カラー熱転写プリンタ **シャープMZ-1P22**

好評発売中！

〈24×24ドット漢字●7色カラー●漢字30字／秒高速印字●MZ1P17とフルコンパチ●5KBのバッハメモリ付〉適応パソコン：MZ2000、2500、5500、6500シリーズ、X1シリーズ、X68000シリーズ他。

標準価格¥59,800➡**特価¥38,640**（ケーブル付）

パソコンファクスMZ-1V01

"プリンタ・コピー・ファクス" 1台3役のスグレモノ 限定セット販売！

●MZ25セット（インターフェースソフト付）
標準価格合計¥342,800➡**¥168,000**

●MZ-1V01（本体のみ）
標準価格合計¥278,000➡**¥98,000**

シャープMZ-1X30 モデムホン

（1×19上位機種）

〈300/1200BPS全2重通信対応モデム内蔵●音声入出力端子付●ダイヤルパルス／プッシュボタン対応●プッシュボタン音解析機能●シャープ手順、CCITT、V25bis通信手順サポート〉

標準価格¥98,000➡**特価¥39,800**

パソコンと専用ワープロをひとつにした16ビット シャープMZ-2861

ワープロソフト「書院28」
MS-DOS V3、1装備
エミュレーションソフト搭載

定価¥328,000➡
大特価¥198,000（ディスプレイは別）

MZ-2861用ソフト（UPシリーズ）

●1P-1251（デスクアップ）……定価¥88,000⇨特価¥20,000
●1P-1252（チャートアップ）……定価¥55,000⇨特価¥20,000
●1P-1253（クリッパー）……定価¥77,000⇨特価¥20,000
●1P-1254（プランナップ）……定価¥88,000⇨特価¥20,000

シャープMZ-2520

定価¥159,800➡**大特価¥80,000**

新発売！《限定発売》

'89プログラム大賞グランプリ受賞作「HEAVY METAL」搭載

PC-E500PJ
定価¥28,800
➡大特価

●ご購入の方にもれなく「ポケコンジャーナル特別号」を進呈。

PC-500と各種パソコンをつなぐインターフェースケーブル
CE-140T　¥8,800

アイビット推奨ディスプレイ

●シャープCZ-830D・BK
（14型）
2モードオートスキャン方式
（アナログ/デジタル）
定価¥98,000⇨
特価¥54,800

CZ-830D対応パソコン機種：CZ880C/881C。X1/TURBOシリーズ。ケーブルは本体付属を使用。NEC PC-8801・9801シリーズ（XA・XLのみ不可）MZ700/1500/2000/2200/2500各シリーズ（推奨品シャープ8D8K）。

●シャープCZ-602D-GY・BK
（15型カラーディスプレイTV）
ドットピッチ3.9
定価¥98,000⇨
特価¥79,000

CZ-602D対応パソコン機種：※X1シリーズ/※X1 turboシリーズ/X1 yurboZシリーズ/X68000シリーズ/PC8801シリーズ/PC-9801シリーズ/PC-286シリーズ
（※は接続ケーブルAN1506が必要です）

拡張機器他

●シャープCZ6BM1（X68用MIDIボード）¥26,800⇨¥23,000
●シャープCZ-8GR(X1.GRAM)‥¥32,000⇨¥12,000
●シャープCZ-8EB3(I/Oボックス)‥¥33,800⇨¥28,000
●シャープCZ-8BK3…(X1)…¥13,800⇨¥11,700
●シャープCZ-8BK4…(X1)……¥6,800⇨¥5,700
●シャープCZ-8BGR2・(X1)……¥14,800⇨¥4,000
●シャープCZ-8BS1…(X1)…¥23,800⇨¥19,500
●シャープCZ-64H（ハードディスク）〈CZ-602・652内蔵用〉……特価
●シャープCZ-8NJ2（インテリジェントコントローラー）¥23,800⇨¥18,500
●シャープCZ-81Tチルトスタンド……¥8,500⇨¥1,000
●シャープMZ-1U08（1500,700拡張）‥¥25,000⇨¥12,000
●シャープMZ-1U03（1500,700拡張）‥¥35,000⇨¥15,000
●シャープMZ-1X22モデムユニット…¥21,800⇨¥13,000
●シャープMZ-1R12 RAM……¥35,000⇨¥8,000
●シャープMZ-1E29 (MZ)……¥17,800⇨¥9,800
●シャープMZ-1E30（2500ハードディスクインターフェイス）…¥25,000⇨¥22,500
●シャープMZ-1U09…(2500)…¥9,000⇨¥7,200
●シャープMZ-1M03…(5500)・¥69,000⇨¥35,000
●シャープMZ8BC04‥(2000)‥¥18,000⇨¥8,000
●シャープMZ-8B104‥(2000)・¥45,000⇨¥18,000
●シャープMZ-1R11…(5500)・¥80,000⇨¥30,000
●シャープMZ-1R24…(1500)‥¥22,000⇨¥6,000
●シャープMZ-1R26A‥(2500)・¥13,000⇨¥12,800
●シャープMZ-1R27A・(2500)・¥13,000⇨¥10,000
●シャープMZ-1R28A・(2500) ¥13,000⇨¥10,000
●シャープMZ-1R29A‥(2500)・¥32,000⇨¥10,000
●シャープMZ-1T02…(2200)‥¥19,800⇨¥8,500
●シャープMZ-1T03…(1500)‥¥12,000⇨¥8,500
●シャープMZ-1X29……¥13,800⇨¥11,000
●テレシステムRM-25E（2500RAMファイル）¥428,800⇨¥38,500
●シャープCZ-600、800チルト台……特価¥3,500
●シャープCZ-8BE2X1 320KRAM……¥29,800⇨¥25,300
●シャープCZ-8BM2X1 232Cマウスボード‥¥19,800⇨¥16,800
●シャープX1、MZ用マウス……特価¥4,800
●シャープX1用ジョイカード……¥1,500
●富士通16βキーボード（親指）・¥25,000⇨¥20,000
●シャープMZ-3500キーボード……¥8,000
●シャープMZ-5500キーボード……¥8,000
●シャープ2000/2200キーボード……¥8,000

(MZ-2861)

●シャープMZ1R35（1MB増設メモリーボード）‥¥55,000⇨¥19,000
●シャープMZ1R36（MZ-2861増設1MRAM）¥45,000⇨¥15,000
●シャープMZ1E26（ボイスコミュニケーション）…¥24,800⇨¥13,000
●シャープSS-SC28M（ハンディーコピーキット）¥49,800⇨¥10,000
●シャープ1E35(ADPCMボード)……¥49,800⇨¥13,000
●シャープ1E39(RE232C 2CHボード)…¥39,800⇨¥13,000

プリンター

●シャープCZ-8PC3……¥65,800⇨¥45,000
●シャープCZ-8PC4（黒・グレー）・¥99,800⇨大特価
●シャープMZ-1P27……¥268,000⇨¥214,400
●シャープMZ-1P28……¥148,000⇨¥118,400
●シャープMZ-1P29……¥168,000⇨¥134,400

フロッピーディスク

●シャープCZ501H（X68拡張用10M）・¥258,000⇨¥60,000
●シャープCZ-503F……¥49,800⇨¥30,000
●シャープCZ-502F……¥99,800⇨¥60,000
●シャープCZ-520F……¥118,000⇨¥70,000
●シャープCZ-53F……¥19,800⇨¥9,800
●シャープCZ-300F(CZ-3PCM付)……¥13,000
●シャープCZ8PG1……¥130,000⇨¥100,000
●シャープCZ8PG2……¥160,000⇨¥130,000

ディスプレイ

●富士通FMTV-153……¥108,000⇨¥76,000
●シャープMZ-1D27……¥120,000⇨¥79,800

ソフト

(MZ-2500用)

●1P-1213 FORTRAN……¥13,800⇨¥11,700
●1P-1215 COBOL……¥13,800⇨¥11,700
●1P-1217 PROLOG……¥11,300⇨¥11,700
●MZ-6Z001 2500 PCPM……¥16,800⇨¥14,200
●DANGER BOX……¥5,800⇨¥2,000
●EXTRA HYPER DISK MONITOR……¥10,000⇨¥8,500
●EXTRA HYPER DISK MONITOR+…¥14,000⇨¥12,000
●FILE UTILITY<UT-25F>……¥6,800⇨¥6,000
●FREE CALL……¥6,800⇨¥1,000
●G-EDIT2500……¥8,000⇨¥7,000
●H.S-コントローラー……¥9,600⇨¥8,500
●HuCAL日本語……¥45,000⇨¥15,000
●SOUND GAL……品切
●アビス2……品切
●ウィザードリィ……品切
●エキサイトバイク……¥6,800⇨¥2,000
●カレイドスコープ……¥9,800⇨¥3,000
●カレイドスコープ2……¥5,800⇨¥1,000
●ザ・ブラックオニキス……¥7,800⇨¥3,000
●スーパー修理屋さん……¥12,000⇨¥10,200
●トップ マネジメント……¥19,800⇨¥6,500
●バルーンファイト……¥6,800⇨¥2,000
●マーベラス……品切
●ムーンチャイルド……¥7,800⇨¥3,000
●英雄伝説サーガ……¥9,800⇨¥2,000
●五目並べ……¥4,800⇨¥2,000
●探検隊第2弾……¥7,800⇨¥2,000
●プリントSHOP……¥9,800⇨¥8,500
●プリントSHOPライブラリー1……¥4,500⇨¥3,800
●プリントSHOPライブラリー2……¥4,500⇨¥3,800

(X1用)

●日本語ワープロ侍 X1t……¥19,800⇨¥16,800
●CZ-8WB51 X1tディスクBASIC……¥9,800⇨¥3,500
●3CP/M X1 3" CPM……¥16,800⇨¥5,000
●CZ-8BK3 X1t第二水準ROM…¥13,800⇨¥11,700
●CZ-128SF X1.CP/M……¥13,800⇨¥11,700
●CZ-115LF X1 FORTRAN……¥13,800⇨¥11,500
●CZ-116LF X1.C……¥13,800⇨¥11,700
●CZ-117SF X1t LOGO……¥18,800⇨¥13,200
●CZ-118LF X1.COBOL……¥13,800⇨¥11,700
●CZ-126LF X1 APL……¥13,800⇨¥11,700
●CZ-130SF X1t CP/M……¥14,800⇨¥12,500
●CZ-134SF X1 LOGO……¥9,800⇨¥8,700
●CZ-137SF X1t ZSSTAFF……¥19,800⇨¥16,800
●CZ-138SF X1 ZSSTAFF……¥13,800⇨¥11,700

(MZ-5500、6500SOFT)

●MZ-2Z013(MZ-5500MSDOS)
●MZ-2Z014(MZ-5500TODAY)
●MZ-2Z023(MZ-5500GW.BASIC)
●MZ-2Z028(MZ-6500GW.BASIC)
●MZ-2Z025(MZ-5500ワープロ)
●MZ-2Z029(MZ-6500TODAY)

本体

●シャープCZ-820、822、880、881、MZ-3500、2520、2861、X68000、CZ-612、662、602、652●富士通FM-77AV-1、77AV-2、77AV20、77AV40 ●NEC PC-9801N ●東芝J3100SS

《全商品新品完全保証付》

■シャープポケコン全商品販売中。カタログ、特価表ご請求ください（〒72）。

☎0426-45-3001～3

FAX.0426-44-6002

●営業時間/10:00～19:00●電話受付/20:00迄可●定休日/日曜日（祭日営業）

SHARP SUPER XEX SHOP

アイビット電子株式会社　〒192　東京都八王子市北野町560-5

上記の広告商品はすべて店頭販売もしております。

全国通販 価売
北海道から沖縄まで

★送料はご注文の際にお問い合わせ下さい。
★掲載の商品は、すべて新品、保証書付きです。
★掲載の商品は充分用意してありますが、ご注文の際は、在庫の確認の上、現金書留または、銀行振込でお申し込み下さい。全商品クレジットでも扱っております。
★お申し込みの際は必ず電話番号を明記して下さい。
★商品、品切れの節はご容赦下さい。

富士銀行八王子支店　（普）1752505

●本誌発売時には上記価格よりさらにお求めやすい価格に変更されている場合があります。●一部を除き上記商品価格には消費税は含まれておりません。全ての商品に対し別途3%の消費税がかかりますのでご了承ください。

ACCESS

X1エミュレータ

好評発売中

定価¥9,800

X1エミュレータはX68000上でX1シリーズのアプリケーションを実行するためのソフトエミュレータです。X1のアプリケーションを完全にソフトウェアのみでエミュレートしているため、X1上での実行速度と比較して、平均3～5倍程度おそくなりますが、X68000のマシン上に実現した仮想X1マシンを楽しめます。また、X1とX68000の相互間でファイルを転送するためのユーティリティと専用ケーブルが付属しますので、X1上で作り上げたソフトの資産をX68000上に移行することも簡単にできます。

X1エミュレータの機能

- X1エミュレータはX1に相当する機能をエミュレート。この仮想コンピュータには最大4つのドライブが仮想的に接続。
- X1エミュレータからみたドライブはHuman68kのドライブ上にあるファイルで仮想的に実現。このファイルはX1用の5" 2Dディスクのイメージをファイル転送ユーティリティでまるごと転送したもの。
- X1エミュレータで仮想的に実現したX1は仮想ドライブから起動。このため仮想ドライブ用ファイルには、X1を立ち上げるために必要なHuBASICやCP/Mなどのシステムプログラムが必要。
- X1エミュレータでは、X1の持つVRAMを含むメモリイメージやZ80CPUを仮想的にソフトウェアで実現。

ファイル転送ユーティリティ

ディスク転送

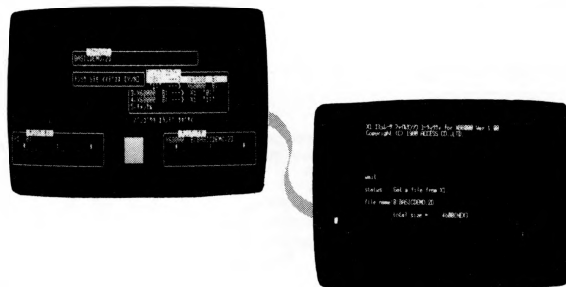
X1ディスク⟷X68000 Human68k(5" 2Dディスクイメージファイル)

- X1エミュレータではHuman68k上のディスクイメージファイルを仮想ドライブとして使用。

ファイル転送

X1 BASIC:CP/M⟷X68000 Human68k

- X1で作ったプログラム&データをX68000上で使用。

※付属の専用ケーブルをX1とX68000に接続してファイルを転送します。

X1エミュレータQ&A

Q. ファイル転送のために別途RS-232Cケーブルを買わないといけないのですか?

A. 専用のケーブルが付属しますのでその必要はありません。

Q. X1BASICのプログラムをX68000上のX-BASICで使えますか?

A. 通常のセーブではコードが違うので使用できませんが、アスキーセーブしたファイルであればX-BASIC上でそのままロード可能です。

Q. TurboBASICで作成した住所録などの漢字を含んだデータがあるのですがX68000上にファイル転送できますか?

A. X1TurboもX68000も漢字はシフトJISコードなのでファイルの転送は可能です。ただし、漢字ROMを必要とするものはサポートしていません。

Q. Turbo用のソフトは動きますか?

A. X1用のみでTurbo専用のソフトは動きません。

Q. ゲームは動きますか?

A. 純粋にBASICでかかれたものは動きますが、プロテクトがかかったものや直接ハードをアクセスするような市販のゲームは動きません。

*タイミング等ハードウェアに依存するようなソフトは、原理上実行できない、もしくは正常に動作しない場合がありますのでご注意ください。
*一部サポートしていない機能があります。

X1エミュレータ通信販売 購入希望として住所、氏名、電話番号をお知らせください。注文書をお送り致します。

発売中　X68000用　**CONCERTO-X68K**　MS-DOSエミュレータ　定価¥99,800

代理店募集

アクセスではこれらの製品の発売にあたり代理店を募集しております。詳しくはお問い合せください。

*この商品価格には消費税は含まれておりません。
*MS-DOSはマイクロソフト社,CP/Mはデジタルリサーチ社の商標です。
文中のソフトウェアは各社の商標です。
*製品の仕様、名称は予告なく変更する場合もございますのであらかじめご了承ください。

有限会社アクセス
〒101 東京都千代田区神田神保町1-64
神保町協和ビル7F
☎03(233)0200(代) FAX.03(291)7019

資料請求券
oh!X
5月号

学生の特権は
無限大ネットワーク。

データ収集/X-MODEM

レポート提出日が迫ってるけど、作業がはかどらない。助けを求めると、いろんな人が資料を送ってくれた。X−MODEMだと、グラフもそのまま送ってもらえるので大助かり。ボクも"おかえし"しなくては！

夏でもスキー/SIG

スキー大好き!!のボクは１年中スキーと離れられない。ＳＩＧにはそんな仲間がいっぱい。シーズン中のスキーツアーはもちろん、オフにもあれこれ情報交換しながら、熱い思いを語り合っている。

家族と交信/電子メール

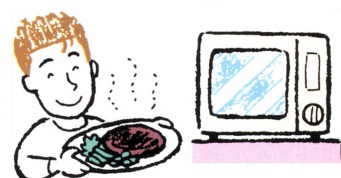
ふるさとの弟は高校生。ボクと同じ大学をめざしてるので、次々と電子メールで情報をきいてくる。返事と一緒に近況をメールしてたら、おふくろや親父もメールの仲間入り。家族でワイワイ交換日記。

……

人。

同郷会/OLT（チャット機能）

ＢＢＳやＳＩＧで呼びかけて同郷の人を募り、待ち合わせてＯＬＴ（オンライントーク）。なつかしい故郷の話題もさることながら、思わぬ人との出会いがいっぱいあって、充実！同郷のよしみっていいなあ。

自炊の味方/データベース

はじめての自炊。安くて簡単で、しかも栄養のあるものを！そこで大活躍するのが、思いきって買った電子レンジ。データベースの"電子レンジ教室"で、レパートリーはぐんぐん広がる。

僕は一人暮らしの大学3回生。J&P HOT LINEでキャンパスライフプラスαをエンジョイ中。

アクセスポイント新設のお知らせ

東京・大阪・名古屋2400bpsを始め、富山・大津・津・堺・熊本1200bpsアクセスポイントを新設しました。全国どこからでも、ますます利用しやすくなったJ&P HOT LINE。あなたも右記のスタータキットで仲間入りしませんか？

J&P HOT LINEは全国90ヵ所のアクセスポイント。2万5千人の仲間が、あなたの仲間になってくれます。

名	
〒	
電	

パソコン/ワープロ通信ネットワークサービス

J&P HOT LINE

アクセスポイントは全国に90ヵ所。日本全国を網羅する、本格的な通信ネットワークです。

スタータキットのお求めはJ&P各店でどうぞ。

店名	住所	電話
渋谷店	東京都渋谷区道玄坂2丁目28番4号	☎(03) 496-4141
町田店	東京都町田市森野1丁目39番16号	☎(0427)23-1313
八王子店	東京都八王子市旭町1番1号八王子そごう7F	☎(0426)26-4141
立川店	東京都立川市幸町4－39－1	☎(0425)36-4141
本厚木店	厚木市中町3－4－3	☎(0462)25-1548
富山店（4月27日金オープン!!）	富山市桜町2－1－10	☎(0764)32-3133
金沢店	金沢市入江2－63	☎(0762)91-1130
寺地店	金沢市寺地2－3	☎(0762)47-2524
大須店	名古屋市中区大須4丁目2-48	☎(052)262-1141
テクノランド	大阪市浪速区日本橋5丁目6番7号	☎(06) 634-1211
メディアランド	大阪市浪速区日本橋5丁目8番26号	☎(06) 634-1511
コスモランド	大阪市浪速区難波中2丁目1番17号	☎(06) 634-3111
U.S.LAND	大阪市浪速区日本橋4丁目9番15号	☎(06) 634-1411
ビジネスランド	大阪市北区梅田1-1-3大阪駅前第3ビルB2	☎(06) 348-1881
梅田店	大阪市北区小松原町1-10	☎(06) 362-1141
高槻店	高槻市高槻町11番16号	☎(0726)85-1212
くずは店	枚方市楠葉花園町15番2号	☎(0720)56-8181
千里中央店	豊中市新千里東町1-3-403千里サンタウン4F	☎(06) 834-4141
摂津富田店	高槻市大畑町24－10	☎(0726)93-7521
寝屋川店	寝屋川市緑町4－20	☎(0720)34-1166
藤井寺店	藤井寺市岡2丁目1番33号	☎(0729)38-2111
岸和田店	岸和田市土生町2451－3	☎(0724)37-1021
さんのみや1ばん館	神戸市中央区八幡通3-2-16	☎(078)231-2111
西宮店	兵庫県西宮市河原町5－11	☎(0798)71-1171
姫路店	姫路市東延末1丁目1番住友生命姫路南ビル1F	☎(0792)22-1221
京都寺町店	京都市下京区寺町通仏光寺下ル恵美須之町549	☎(075)341-3571
京都近鉄店	京都市下京区烏丸通七条下ル東塩小路町702	☎(075)341-5769
和歌山店	和歌山市元寺町4丁目4番地	☎(0734)28-1441
奈良1ばん館	奈良市三条町478－1	☎(0742)27-1111
郡山インター店	大和郡山市横田693－1	☎(07435)9-2221
熊本店	熊本市手取本町4－12	☎(096)359-7800

T4910217905561　雑誌02179-5

Oh!X 5月号 1990年5月1日発行(毎月1回1日発行) 通巻第97号 昭和58年11月2日第三種郵便物認可 (株)日本ソフトバンク発行 Printed in Japan 定価560円(本体544円)